CISCO SYSTEMS

Cisco MGX 8850 Routing Switch Command Reference

Release 2.0

October 2001

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-10467=
Text Part Number: 78-10467-04 Rev C0

Cisco Reader Comment Card

General Information

1 Years of networking experience: _______ Years of experience with Cisco products: _______

2 I have these network types: ☐ LAN ☐ Backbone ☐ WAN
☐ Other: ______________________________

3 I have these Cisco products: ☐ Switches ☐ Routers
☐ Other (specify models): ______________________________

4 I perform these types of tasks: ☐ H/W installation and/or maintenance ☐ S/W configuration
☐ Network management ☐ Other: ______________________________

5 I use these types of documentation: ☐ H/W installation ☐ H/W configuration ☐ S/W configuration
☐ Command reference ☐ Quick reference ☐ Release notes ☐ Online help
☐ Other: ______________________________

6 I access this information through: ____% Cisco.com (CCO) ____% CD-ROM
____% Printed docs ____% Other: ______________________________

7 I prefer this access method: ______________________________

8 I use the following three product features the most:

Document Information

Document Title: Cisco MGX 8850 Routing Switch Command Reference

Part Number: DOC-7810467= S/W Release (if applicable): Release 2.0

On a scale of 1–5 (5 being the best), please let us know how we rate in the following areas:

_____ The document is written at my technical level of understanding.

_____ The information is accurate.

_____ The document is complete.

_____ The information I wanted was easy to find.

_____ The information is well organized.

_____ The information I found was useful to my job.

Please comment on our lowest scores:

Mailing Information

Company Name Date

Contact Name Job Title

Mailing Address

City State/Province ZIP/Postal Code

Country Phone (   ) Extension

Fax (   ) E-mail

Can we contact you further concerning our documentation? ☐ Yes ☐ No

You can also send us your comments by e-mail to **bug-doc@cisco.com**, or by fax to **408-527-8089**.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 4631 SAN JOSE CA

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN DOCUMENT RESOURCE CONNECTION
CISCO SYSTEMS INC
170 WEST TASMAN DRIVE
SAN JOSE CA 95134-9883

CONTENTS

CHAPTER 7

FIGURES

TABLES

About This Manual

Welcome to the command line interface (CLI) documentation for the Cisco MGX™ 8850 wide area routing switch, Release 2.

This chapter discusses:

- Objectives
- Audience
- Organization
- Related Documentation
- Conventions
- Command Changes

Cisco documentation and additional literature are available in a CD-ROM package, which ships with your product. The Documentation CD-ROM, a member of the Cisco Connection Family, is updated monthly. Therefore, it might be more current than printed documentation. To order additional copies of the Documentation CD-ROM, contact your local sales representative or call customer service. The CD-ROM package is available as a single package or as an annual subscription. You can also access Cisco documentation on the World Wide Web at http://www.cisco.com, http://www-china.cisco.com, or http://www-europe.cisco.com.

If you are reading Cisco product documentation on the World Wide Web, you can submit comments electronically. Click **Feedback** in the toolbar, select **Documentation**, and click **Enter the feedback form**. After you complete the form, click **Submit** to send it to Cisco. We appreciate your comments.

Objectives

This publication provides instructions for using the MGX 8850 commands in the CLI.

Audience

The Cisco command line interface lets you control the network from a level somewhat below that provided by Cisco WAN Manager (formerly Cisco StrataView Plus). This document helps network designers and operators to set up, manage, and troubleshoot networks.

Organization

The chapters in this guide are:

Chapter 1 **Command Line Interface Overview**

Describes the how to use the command line interface CLI and includes descriptions of user-oriented commands.

Chapter 2 **Shelf Management Commands**

Describes commands that pertain primarily to switch-level configuration.

Chapter 3 **Equipment and Resource Provisioning**

Describes commands for specifying physical characteristics to the node and logical resources to the controllers.

Chapter 4 **ILMI Commands**

Describes commands that add, modify, display status, and create statistics for ILMI.

Chapter 5 **PNNI Commands**

Describes commands to add, delete, configure, display status, and create statistics for the PNNI routing protocol.

Chapter 6 **Logical Node, Port, and Signaling Commands**

Describes commands to specify node-level and PNNI logical port parameters.

Chapter 7 **SPVC and SVC Commands**

Describes commands to add, delete, configure, display, and specify statistics for SPVCs.

Chapter 8 **Network Optimization Commands**

Describes commands to optimize PNNI routing.

Chapter 9 **Troubleshooting Commands**

Describes commands that help you troubleshoot the node and the network.

Index **Index**

Lists the name and chapter and page number for every command in the manual.

Related Documentation

This section lists documentation that applies to the MGX 8850 Release 2 switch and associated products in a Cisco WAN. Table 1 lists the product documentation for the MGX 8850 Release 2 switch.

Table 1 MGX 8850 Switch Release 2 Related Documentation

Documentation	Description
Cisco MGX 8850 Hardware Installation, Release 2 DOC-7810351=	Provides a detailed description for installing the MGX 8850 switch in a restricted access location.
Cisco MGX 8850 Routing Switch Command Reference, Release 2 DOC-7810467=	Describes and lists the user-accessible CLI[1] for the MGX 8850 ATM edge switch.
Cisco MGX 8850 Switch Software Configuration Guide, Release 2 DOC-7810352=	Describes how to configure the MGX 8850 to operate as an ATM core switch or as an ATM edge switch.
Cisco PXM SNMP Reference, Release 2 DOC-7811276=	Provides information on all supported MIB[2] objects, support restrictions, traps, and alarms for the PXM45 Module.
Cisco AXSM SNMP Reference, Release 2 DOC-7811369=	Provides information on all supported MIB objects, support restrictions, traps, and alarms for the AXSM Module.
Cisco PNNI SNMP Reference, Release 2 DOC-7811277=	Provides information on all supported MIB objects, support restrictions, traps, and alarms for the PNNI Module.

1. CLI = command line interface
2. MIB = Management Information Base

Table 2 lists the documentation for the CiscoView product running on an MGX 8850 Release 2 node.

Table 2 WAN CiscoView for MGX 8850 Release 2 Related Documentation

Documentation	Description
WAN CiscoView for Release 2 of the MGX 8850 *DOC-7810349=*	Provides instructions for using WAN CiscoView, a management application that lets you configure and troubleshoot equipment.

Table 3 lists the documentation for the Cisco WAN Manager (CWM) network management system.

Table 3 Cisco WAN Manager Release 10 Related Documentation

Documentation	Description
Cisco WAN Manager Installation for Solaris, Release 10 DOC-7810308=	Provides procedures for installing Release 10 of the CWM network management system on Solaris systems.
Cisco WAN Manager User's Guide, Release 10 *DOC*-7810658=	Provides procedures for operating Release 10 of the CWM network management system.
Cisco WAN Manager SNMP Service Agent Guide, Release 10 DOC-7810786=	Provides information about the CWM Simple Network Management Protocol Service Agent components and capabilities.
Cisco WAN Manager Database Interface Guide, Release 10 DOC-7810785=	Provides the information to gain access to the CWM Informix OnLine database that is used to store information about a network.

Conventions

This publication uses the following conventions to convey instructions and information.

Command descriptions use these conventions:

- Commands and keywords are in **boldface**.
- Arguments for which you supply values are in *italics*.
- Elements in square brackets ([ ]) are optional.
- Alternative but required keywords are grouped in braces ({ }) and are separated by vertical bars (|).

Examples use these conventions:

- Terminal sessions and information the system displays are in `screen` font.
- Information you enter is in **`boldface screen`** font.
- Nonprinting characters, such as passwords, are in angle brackets (< >).
- Default responses to system prompts are in square brackets ([ ]).

Note Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Command Changes

This section contains lists of commands that are either new, changed, or removed in this release.

New Commands

These commands are new to release 2:

- abortallsaves
- actaudit
- addlnloop
- addlpback
- ccc
- checkflash
- clralm
- clralmcnt
- clrbecnt
- clrchancnts
- clrcnf
- clrloginmsg
- clrportcnts
- clrqosdefault
- cnfcbclk
- cnfloginmsg
- cnfpasswd
- cnfpswdreset or cnfpasswdreset
- conntrace
- core
- dbgcon
- dellpback
- delpnni-node
- delpnni-summary-addr
- delsesn
- dspactaudit
- dspapsbkplane
- dspbecnt
- dspbkpl
- dspcbclk
- dspconfigs

- dspconload
- dspdbinfo
- dsperrs
- dsplnbucketcnt
- dsplnload
- dsploginmsg
- dsplpback
- dsppatterns
- dsppnni-dbg
- dsppnni-election
- dspportload
- dsppswdreset or dsppasswdreset
- dsprevs
- dspsesn
- dspsscopstats
- dspsvcparm
- smgrDataShow
- softswitch
- switchback
- telnet
- uplmi
- users

Changed Commands

The commands that changed in this release by being executed on a different board, by requiring different parameters, or by having different keywords are not listed here. See individual commands for details.

The following commands were replaced in this release:

- dspspvclog was renamed dspcons-dbg
- passwd was renamed cnfpasswd
- dsppwd was renamed dsppasswdreset
- cnfspvclog was renamed dbgcon

Removed Commands

The following commands have been removed:

- addln—removed from AXSM
- addmaster (obsolete)

- addslave (obsolete)
- clrxbaralms
- cnfifip (obsolete)
- cnfxbaradmin (obsolete)
- delln—removed from AXSM
- dspifip (obsolete)
- dspnddebug
- dspshelfalm (obsolete)—pre-empted by dspenvalms
- dspslotalms
- dspxbaralms (removed from 2.0)—the singular command, dspxbaralm, remains
- formatdisk
- mkfs
- offdiagcstat
- shutdisk
- syncdisk

Command Line Interface Overview

This chapter describes the command line interface (CLI) for the Cisco MGX 8850 node. In addition, the chapter describes the basic user-commands for logging on to and out of the switch, changing between the CLI of different cards, and listing files on the hard drive. This chapter includes usage examples. For information on how to configure a switch and basic network services, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0..

The chapter describes:

- The role of the CLI on the switch
- The information contained in the CLI prompt
- The command syntax
- Contents of a command description
- Identification of the models of the ATM Switching Service Module (AXSM)
- A logical port in the context of the Public Network-to-Network (PNNI) protocol
- A logical port in the context of AXSM configuration
- User-specific commands

The Role of the CLI

During normal operation, the tools for configuring, monitoring, and controlling a switch are the CiscoView application for equipment management and the Cisco WAN Manager application for connection management. However, during initial switch installation, troubleshooting, or where low-level control is important, the command line interface (CLI) provides the best access to the switch.

Each PXM45 and service module supports its own CLI. Each card has a set of commands specific to its card-type—although some commands overlap. In certain instances, this manual indicates cases where two commands with the same name produce different results on different card types. (The available command set also depends on the privilege level of the user.)

Although you can execute a command on only the card that supports that command, the target of the command can be another card when you are “on” the PXM45. (Being “on” a card means you have logged into the card and are using the CLI for that card.) On the PXM45, you can execute commands that target the PXM45 itself, the whole node, or a service module.

To move from the CLI of one card to the CLI of another card, use the Change Card (**cc**) command. The description of **cc** appears in this chapter.

Command Line Prompt

The format of the CLI prompt is:

name.slot number.card type.card state >

where:

- *name* is the name of the node (“Unknown” until a you assign name with the **cnfname** command).
- *slot number* is the slot of the front card.
- *card type* identifies the Processor Switching Module 45 (PXM45) or a service module type, such as the AXSM.
- *card state* is “i” for initialized, “a” for active, or “s” for standby. For many commands, a card must be active before the command can execute, so each command description has an Attributes section that identifies the states in which the command can execute.
  - A card in the initialized state (i) is still loading application modules.
  - A card in the active (a) state either is fully configured and ready to carry out its function or is already performing its function with live traffic.
  - Typically, a card goes into the standby (s) state when it first powers up and boots or when you execute a command that puts it in the standby state. For example, the commands for a graceful upgrade of firmware on a pair of PXM45s puts the active card in the standby state and the standby card in the active state (see **loadrev** description for details).

An example of a CLI prompt is:

```
MGX8850.7.PXM45.a >
```

The preceding prompt shows that the

- Name of the node is “excel.”
- Slot number is 7.
- Card type is PXM45.
- Card state is active.

Command Syntax

This section contains the following syntax topics:

- Notation
- Position-dependent parameters
- Keyword-driven parameters
- Logical port format
- Command entry

Notation

The notation for command and argument parameters follows:

- Commands and their parameters are separated by a space.
- Variables appear in *italics*.
- Keywords and commands appear in **bold**.
- Required arguments appear within left and right arrowheads (“< >”).
- Optional parameters appear within square brackets (“[ ]”).
- A vertical bar (|) represents the logical OR function.

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Contents of a Command Description

Each command description contains:

- An introductory paragraph that explains the function of the command. Additional paragraphs elaborate on the functionality as needed.
- A list of cards on the CLI of which you can execute the command.

- The syntax of the command. This manual presents parameters in a column to make them easier to read, particularly when displayed through an electronic medium.
- A syntax description lists all the parameters. Each item in the list includes a brief definition, functional details when appropriate, the range of values for the parameter, and an applicable default value.

  Note that, in many instances, the default value is not merely a basic starting value but rather the most desirable or commonly used value.
- Occasionally, the description includes a Usage Guidelines section when the complexity of the command warrants it. The Usage Guidelines section contains important details about using the command. When needed, an additional section with a specialized focus may appear. An example is the Version Numbering description for the firmware upgrade commands (see **loadrev**, for example).
- An Attributes section lists the following details:
- Whether the switch logs each instance of command execution. Typically, the switch logs each configuration change but no display commands.
- The state of the card required to a execute a command. The state can be active, standby, initialized (infrequently), or any of these states.
- A Related Commands lists other commands in the typical grouping of commands (add, delete, configure, and display) or other commands that could complement the command.
- An Example section shows one or more examples of command usage. The text for this section describes the intention of the command and may also describe an outcome. A representation of screen output usually appears. Occasionally, supplemental commands and screen samples appear in support of the example.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 1-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node uses the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation*, Release 2.

Table 1-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–4	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 1-2 and Table 1-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 1-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 1-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay

- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf*.]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 1-4.

Table 1-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 1-4 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1
- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

List of Commands

The commands in this chapter appear in Table 1-5. It shows the name and function of each command.

Table 1-5 Commands

Command	Description
? or Help	Help (list of commands available on the card).
bye	End current user-session.
cc	Change card
cd	Change directory.
clrscrn	Clear terminal screen.
cmdhistory	List last 10 commands entered.
copy	Copy one file to another on the hard drive.
cp	Copy one file to another on the hard drive.
del	Delete a file on the hard drive.
exit	Exit the current user-session (and log out).
help or ?	List of commands per card.
history	List recent command entries.
logout	Log out of the node (and end current user-session).
ls	List files on the hard drive.
ping	Send ICMP packet to far end station to determine if it is operational.
pwd	Identify current working directory on the hard drive.
who	List current user on the PXM45.
whoami	List details about the current user.

?

Help

Use **help** to view all commands you can execute on the current card and at the current privilege level. The display does not show commands with a privilege level that is higher than that of the current user.

If you follow the **?** with part of a command name, the output shows all commands that contain that string. If you follow the **?** with the complete name of one command, the output simply states whether that command is available.

If you can enter two parameter strings, **help** provides information for each of the two strings separately (not a single, two-part string).

Cards on Which This Command Runs

PXM45, AXSM

Syntax

?
[*command*]

Syntax Description

command	Full or partial name of a command.

Related Commands

help

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Examples

View all commands associated with the partial command string "con."

```
MGX8850.1.AXSM.a >? con

Available commands
------------------
addcon
clrconcnt
cnfcon
delcon
delcons
dspcon
dspconcnt
dspcons
```

bye

Bye

Exit the current CLI session.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

bye

Related Commands

logout, **exit**

Attributes

Log: log	State: active, standby, init	Privilege: ANYUSER

Exit the current CLI shell.

```
MGX8850.8.PXM.a > bye

(session ended)
```

cc

Change Card

Use **cc** to change from the current CLI to the CLI of another card. Follow **cc** with a slot number.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

`cc` *`<slot number>`*

Syntax Description

slot number	The number of the destination card slot.

Related Commands

None

Attributes

Log: log	State: active, standby, init	Privilege: ANYUSER

Change from the command line of the AXSM in slot 12 to the command line of the PXM45 in slot 8.

```
MGX8850.12.AXSM.a > cc 8

(session redirected)

MGX8850.8.PXM.a >
```

If the slot is empty or the card is unreachable, the system displays an applicable message.

cd

Change Directory

Use **cd** to change to another directory on the PXM45 hard disk.

Cards on Which This Command Runs

PXM45

Syntax

cd *<directory_name>*

Syntax Description

directory_name	Name of the destination directory.

Related Commands

ls, **pwd**, **rename**, **rm/rmdir**, **copy**

Attributes

Log: log	State: active, standby, init	Privilege: ANYUSE

Change directory to FW, then check the result by executing pwd.

```
MGX8850.7.PXM.a > cd FW

MGX8850.7.PXM.a >
```

Verify the current directory by using the **pwd** command.

```
MGX8850.7.PXM.a > pwd
C:FW
```

Go back to Root directory, then check the result by executing pwd.

```
MGX8850.7.PXM.a > cd ..

MGX8850.7.PXM.a > pwd
C:
```

clrscrn

Clear Screen

The **clrscrn** command clears the control terminal screen. After this command runs, only the current command line prompt appears on the screen.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
clrscrn
```

Related Commands

None

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Clear the screen.

```
MGX8850.11.AXSM.a > clrscrn
MGX8850.11.AXSM.a >
```

cmdhistory

Display Command History

The **cmdhistory** command lists the last 10 commands executed on the current card. To execute a previous command with parameters, type an exclamation mark and the associated number and no spaces, then press Enter or Return.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

cmdhistory

Syntax Description

This command takes no parameters.

Related Commands

history

Attributes

Log: no log   State: active, standby, init   Privilege: ANYUSER

Example

Display the previous commands executed on the PXM45. Repeat **dspcd**.

```
MGX8850.7.PXM.a > cmdhistory
Size of cmdHistory is currently 10 line(s)
  1 cmdhistory
  2 cd
  3 cd ..
  4 pwd
  5 cd ..
  6 dspcd
  7 clrscrn
  8 ?
  9 q
 10 cmdhistory


MGX8850.7.PXM.a > !6
dspcd
MGX8850                            System Rev: 02.00   Sep. 09, 2000 16:58:23 GMT
MGX8850 (MGX8850)                                      Node Alarm: MAJOR
Slot Number    7    Redundant Slot:  8

                    Front Card          Upper Card          Lower Card
                    ----------          ----------          ----------

Inserted Card:      PXM45               UI Stratum3         PXM HardDiskDrive
```

```
Reserved Card:      PXM45               UI Stratum3         PXM HardDiskDrive
State:              Active              Active              Active
Serial Number:      SAK0401006C         SHELFMGMGRP         SAK03520058
Prim SW Rev:        2.0(1)D             ---                 ---
Sec SW Rev:         2.0(1)D             ---                 ---
Cur SW Rev:         2.0(1)D             ---                 ---
Boot FW Rev:        2.0(233)A1          ---                 ---
800-level Rev:      16                  B0                  05
Orderable Part#:    800-06147-01        800-03145-07        800-05052-03
CLEI Code:          0000000000          CISCO__INC          0000000000
Reset Reason:       On Reset From Shell
Card Alarm:         NONE
Failed Reason:      None
Miscellaneous Information:
```

copy

Copy

Use **copy** to copy a file to a new file on the disk on the PXM45-HD. This command is the same as the **cp** command.

Cards on Which This Command Runs

PXM45

Syntax

`copy` *`<source file name> <destination file name>`*

Syntax Description

source file name	The name of the file you intend to copy.
destination file name	The name of the new file resulting from **copy** or the name of the existing file that is over-written as a result of **copy**.

Related Commands

cp, **cd**, **ls, rm**, **pwd**, **rename**

Attributes

Log: log	State: active, standby, init	Privilege: GROUP1

Example

Create a new firmware file without the image's suffix by copying the file named pxm_1.0.00Ef.fw to pxm_1.0.00.fw.

```
MGX8850.8.PXM.a > copy pxm_1.0.00Ef.fw pxm_1.0.00.fw
MGX8850.8.PXM.a >
```

cp

Copy

Use **cp** to copy a file to a new file on the disk on the PXM45-HD. This command is the same as the **copy** command.

Cards on Which This Command Runs

PXM45

Syntax

cp *<source file name> <destination file name>*

Syntax Description

source file name	The name of the file you intend to copy.
destination file name	The name of the new file resulting from **cp** or the name of the existing file that is over-written as a result of **cp**.

Related Commands

cd, **ls, rm**, **pwd**, **rename**

Attributes

Log: log	State: active, standby, init	Privilege: GROUP1

Example

Create a new firmware file without the image's suffix by copying the file named pxm_1.0.00Ef.fw to pxm_1.0.00.fw.

```
MGX8850.8.PXM.a > cp pxm_1.0.00Ef.fw pxm_1.0.00.fw
```

del

Delete

Use **del** to remove a file or directory from the PXM45 hard drive.

Cards on Which This Command Runs

PXM45

Syntax

```
del <path_name>
```

Syntax Description

path_name	Name of an existing file or directory.

Related Commands

None

Attributes

Log: log	State: active, standby, init	Privilege: GROUP1

exit

Exit from User Session

Use **exit** to exit the current user session and log out. To start another session, you must log in by using telnet (for example).

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
exit
```

Related Commands

bye, logout

Attributes

Log: log	State: active, standby, init	Privilege: ANYUSER

Example

Exit from the current user session.

```
MGX8850.8.PXM.a > exit
```

```
(session ended)
```

help

Help

Use **help** to view commands associated with the current card. The **help** command is case-sensitive. Its behavior with or without parameters is:

- With no parameter string, it lists all commands on the card.
- With part of a command name, it lists all commands that contain that string.
- With the entire command name, the output shows only whether the command is available.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
Help
[string]
```

Related Commands

?

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

history

Command History

Use **history** to display the last 10 commands executed on the current card. To repeat a command with its parameters, type an exclamation mark followed by the associated number and no spaces.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

history

Syntax Description

This command takes no parameters.

Related Commands

cmdhistory

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the last 10 commands executed on the PXM45. Repeat the previous execution of **dspdate**.

```
MGX8850.7.PXM.a > dspdate
    Sep 09 2000 16:37:37 GMT

MGX8850.7.PXM.a > history
Size of cmdHistory is currently 10 line(s)
  1 q
  2 help dsp
  3 help dspdate
  4 q
  5 help dspdate
  6 history
  7 6
  8 help dspdate
  9 dspdate
 10 history

MGX8850.7.PXM.a > !9
dspdate
    Sep 09 2000 16:38:06 GMT
```

logout

Log Out

The **logout** command lets you end the current user session.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

logout

Syntax Description

This command takes no parameters.

Related Commands

bye, **exit**

Attributes

Log: log	State: active, standby, init	Privilege: ANYUSER

Example

Log out of the current CLI shell.

```
MGX8850.8.PXM.a > logout

(session ended)
```

ls

List

Use **ls** to list the contents of the working directory. The filename is listed for each entry. The total space of the file system and free space is also summarized at the end of the output.

Cards on Which This Command Runs

PXM45

Syntax

ls
[*dir*]

Syntax Description

You can specify an optional directory or path to list.

Related Commands

cd, **pwd**, **rename**, **copy**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Examples

List all the files at the highest level of the disk.

```
MGX8850.7.PXM.a > ls
SM
FW
DIAG
STATS
TMP
CNF
RPM
LOG
clrDB
upgrade.state
config.sys
DB

In the file system :
    total space :  819200 K bytes
    free  space :  700583 K bytes
```

Change to the SCT directory. List all files in the SCT directory, then list the files in the AXSM directory.

```
MGX8850.7.PXM.a > cd /SCT

MGX8850.7.PXM.a > ls
.
```

```
..
AXSM

In the file system :
    total space :  819200 K bytes
    free  space :  660582 K bytes

MGX8850.7.PXM.a > ls AXSM
.
..
AXSM_SCT.CARD.2
AXSM_SCT.CARD.3
AXSM_SCT.PORT.2
AXSM_SCT.PORT.3

In the file system :
    total space :  819200 K bytes
    free  space :  660582 K bytes
MGX8850.7.PXM.a >
```

ping

Ping

Use **ping** to determine if a host is operational. The command causes the switch to send an ICMP packet to a destination address.

Cards on Which This Command Runs

PXM45

Syntax

ping *<IP_Addr>*
[*<Num_Packets>*]

Syntax Description

IP_Addr	IP address of the destination host in dotted decimal format.
Num_Packets	Number of packets. The range is 0–65535. • 0=infinite • 3=default

Related Commands

None

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Ping IP address 172.29.23.148.

```
MGX8850.7.PXM.a > ping 172.29.23.148
PING 172.29.23.148: 56 data bytes
64 bytes from 172.29.23.148: icmp_seq=0. time=0. ms
64 bytes from 172.29.23.148: icmp_seq=1. time=0. ms
64 bytes from 172.29.23.148: icmp_seq=2. time=0. ms
----172.29.23.148 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0
```

pwd

Present Working Directory

Use **pwd** to identify the current working directory on the PXM45.

Cards on Which This Command Runs

PXM45

Syntax

pwd

Syntax Description

This command takes no parameters.

Related Commands

cd, **rmdir**, **rm**, **ls**, **copy**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Identify the present working directory.

```
MGX8850.7.PXM.a > pwd
C:
MGX8850.7.PXM.a >
```

who

Who

Use **who** to see details about the user currently logged into a card. The information consists of the:

- Type of port where you logged into the card
- Slot number of the current card
- Idle time in hours, minutes, and seconds
- Current username
- IP address of the device that accessed the card (not the IP address of the card or node)

Cards on Which This Command Runs

PXM45, AXSM

Syntax

who

Syntax Description

This command takes no parameters.

Related Commands

adduser, **deluser**, **whoami**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display information about the user currently logged into the card.

```
MGX8850.7.PXM.a > who

Port        Slot      Idle       UserId       From
-------------------------------------------------------------
telnet.01 *    7     0:00:00     admin 171.71.25.240
```

whoami

Who Am I

View the current login ID, access level, and associated terminal port of the current user.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
whoami
```

Syntax Description

This command takes no parameters.

Related Commands

adduser, **deluser**, **who**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display information about the user of the current terminal session.

```
MGX8850.7.PXM.a > whoami

User ID:        cisco
Access Level:   CISCO_GP
Terminal Port:  telnet.01

MGX8850.7.PXM.a >
```

CHAPTER 2

Shelf Management Commands

This chapter describes the shelf management commands and some commands that belong to other logical entities running on the switch—PNNI, for example. These commands allow you to add, delete, configure, display status for, and create statistics for node-level features.

The functional areas under shelf management are:

- Node-level parameters of nodename, date, time, time zone, and so on.
- Firmware downloading.
- Network synchronization
- SNMP configuration
- User account creation (and associated password)
- IP connectivity to support user-control of the switch
- Saving and restoring node configuration information

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 2-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node uses the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation, Release 2*.

Table 2-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–4	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 2-2 and Table 2-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 2-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 2-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf*.]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 2-4.

Table 2-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 2-4 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1

- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

abortrev

Abort Revision—return to the previous firmware version.

The **abortrev** command causes the target card to use the previous operational firmware image. It provides a way out of a graceful upgrade that has shown signs of unacceptable performance. (For example, a new feature may not perform as expected.) The commands for changing firmware versions commands run on the PXM45, but they can target either a service module or the PXM45 itself.

You can execute the **abortrev** command after you have executed either **loadrev** or **runrev** but before **commitrev**. (After **commitrev,** the only way to restore the previous version is to force-load it by executing **setrev** or **restoreallcnf**.) The following list outlines the sequence for a graceful upgrade. For a state-by-state view that elaborates on this subject, see Table 2-5 and Table 2-6.

1. **loadrev** loads a firmware version from the hard disk to a card's memory. In a non-redundant card setup, **loadrev** does not cause the system to reset the card.
2. **runrev** causes the primary card to start running the new version. For a redundant pair of cards, the standby becomes the active card then starts running the new version.
3. If an unacceptable problem occurs, the optional **abortrev** command restores the previous version of firmware as well as the previous database contents.
4. **commitrev** declares the new primary version to be acceptable and removes the old primary from main memory (but not the hard disk).

A graceful upgrade takes a single card or a redundant card pair through different states. In addition, the stage at which you execute **abortrev** on a redundant pair determines whether the system resets one or both cards in the pair. The reset depends on whether you execute **abortrev** before or after **runrev**. The stages of a graceful upgrade and the reset actions appear in Table 2-5 and Table 2-6. For a single-card upgrade, see Table 2-5. For a redundant-pair upgrade, see Table 2-6.

The tables start by showing that, initially, the primary and secondary versions of firmware are 2.x, so the only possible operational version is 2.x. The **loadrev** command loads a generic version called 2.y, and the upgrade sequence progressively changes the primary and secondary firmware versions.

Table 2-5 Single-Card Upgrade From 2.x to 2.y

Firmware Status	Initial Version	After loadrev	After runrev	After commitrev
Primary	2.x	2.x	2.y	2.y
Secondary	2.x	2.y	2.x	2.y
Operational	2.x	2.x	2.y	2.y
			After **runrev**, the card resets.	

Note Of special note in Table 2-6, **runrev** causes the standby card to become the active card. The reversed location of the “Active” and “Standby” columns shows the changed states.

Table 2-6 Redundant Pair Upgrade From 2.x to 2.y

Firmware status	Before upgrade		After loadrev		After runrev		After commitrev	
	Active	Standby	Active	Standby	Standby	Active	Standby	Active
Primary	2.x	2.x	2.x	2.x	2.y	2.y	2.y	2.y
Secondary	2.x	2.x	2.y	2.y	2.x	2.x	2.y	2.y
Current	2.x	2.x	2.x	2.y	2.y	2.y	2.y	2.y
			abortrev resets only standby card.		**abortrev** resets both cards.			

Note After you execute **runrev**, the PXM45 updates the database records on disk if changes occur (such as changes to the configuration or network topology). If you revert to the previous version by executing **abortrev**, the post-**runrev** changes are lost. For example, if a switch was added to the network between **runrev** and **abortrev**, the restored database has no record of the topology change.

Cards on Which This Command Runs

PXM45

Syntax

abortrev *<slot> <revision>*

Syntax Description

slot	Number of the slot where firmware must revert to previous version.
revision	Revision number derived from the name of the firmware file. For an explanation, see the section, “Version Numbering Conventions,” in the **loadrev** description.

Related Commands

loadrev, **commitrev**, **runrev**, **setrev**

Attributes

Log: log	State: active, standby	Privilege: SERVICE_GP

Example

Abort the graceful upgrade to firmware file pxm45_002.000.000.000_mgx.fw (so 2.0(0) is the *version*). The system prompts you to confirm that you want the command to execute.

```
pinnacle.8.PXM.a > abortrev 8 2.0(0)
```

addtrapmgr

Add Trap Manager

Set up an SNMP manager that you intend to receive SNMP traps. The maximum number of trap managers on a node is 12.

The trap managers you add through **addtrapmgr** and the trap managers that are added by the SNMP manager (Cisco WAN Manager or other application) do not age and are not deleted. To delete a trap manager, use either the **deltrapmgr** command or an SNMP Set on the intended object.

Cards on Which This Command Runs

PXM45

Syntax

addtrapmgr *<ip_addr>* *<portnum>*

Syntax Description

ip_addr	IP address in dotted decimal format: *nnn.nnn.nnn.nnn*, *n*=0-9 and *nnn* < 256
portnum	Port number on the workstation that receives traps. The range is 0–65535. If you add a trap manager through SNMP, the default *portnum* is 162.

Related Commands

deltrapmgr, **dsptrapmgr**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Add a trap manager with IP address 161.10.144.56 to port 50.

```
node501.7.PXM.a > addtrapmgr 161.10.144.56 50
```

adduser

Add User

Adds a user account with associated name, privilege level, and password. User names must begin with an alpha character. The maximum number of users is 100.

The privilege level of the user you are adding must be lower than the user-level at which you execute **adduser**. For example, to create a user with a privilege 1, you must log in as a superuser or above.

You can execute commands that require either the same or lower level privilege. With superuser access, for example, you can execute commands that require “superuser,” “group 1,” or “anyuser” privilege. The minimum access level for a command appears in the Attributes section of each description.

In *descending* order of access privilege:

- Service (for potentially dangerous configuration commands or complex troubleshooting commands
- Superuser
- Group 1
- Any user

Cards on Which This Command Runs

PXM45

Syntax

```
adduser
<user ID>
<accessLevel>
```

Syntax Description

After you enter a user ID and access level, the system prompts for a password, as the example shows.

user ID	String that you enter to log into the CLI of a PXM45 or a service module. Note that: • The name can consist of up to 12 characters composed of alpha and numeric characters and can include the special characters “_” and “-” but no spaces. • The name must begin with an alpha character. • The name is case sensitive. • The maximum number of user-names on a switch is 50.
accessLevel	System privilege level to be allocated for the user ID. Note that the *accessLevel* is case-sensitive and must be entered as it appears below: • SERVICE_GP • SUPER_GP • GROUP1 • ANYUSER The new user that you configure must have a lower accessLevel than that of the current user.

Related Commands

cnfuser, **dspusers**, **deluser**, **cnfpasswd**, **whoami**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Add a user named "fin" with privilege level GROUP1. To add a GROUP1 user, the current user-prefilter level must be SUPER_GP or higher. To determine the current username, execute the **whoami** command. To see all current privilege levels, execute **dspusers**.

If the privilege level of the current user in this example is GROUP1 or lower, the command fails after you enter the password for the second time, and the system returns a message stating that you entered an incoherent value for "–l " (the level).

```
pinnacle.7.PXM.a > adduser fin GROUP1

Enter password:
Re-enter password:
```

Add the user "leroy" but without establishing a password for "leroy." The system displays the default password "newuser." Subsequently, either a network administrator or the user "leroy" can execute the **cnfpasswd** command to create a password.

```
pop20two.7.PXM.a > adduser leroy ANYUSER
Enter password:
(default password "newuser" will be used)
```

bootChange

Boot Change

Sets the boot IP address and gateway address of the PXM45 card. The boot IP address is used only when the PXM45 card boots up.

In the current release, the only parameters you should enter are “inet on ethernet (e)” and “gateway inet (g).” The **bootChange** command presents one parameter at a time. Therefore, press the Return (or Enter) key at each prompt except for these two. The example in this description shows the two fields where you need to enter an IP address and the fields you skip.

Note The boot IP address does not get saved with **saveallcnf**.

Note Use the **ipifconfig** command to assign IP addresses for the PXM45 and the shelf.

Cards on Which This Command Runs

PXM45

Syntax

`bootChange`

Related Commands

none

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

Specify an IP address of 170.11.52.61 for the Ethernet port and 170.11.52.2 for the gateway IP address. The display shows all the fields that the node presents. For all fields except the ethernet and gateway prompts, press Return or Enter.

```
pinnacle.7.PXM.a > bootChange

'.' = clear field; '-' = go to previous field; ^D = quit

boot device          : lnPci
processor number     : 0
host name            : winter
file name            : /users/joloughl/pxm45_002.000.014-A1.fw
inet on ethernet (e) : 170.11.52.61
inet on backplane (b):
host inet (h)        : 170.11.25.42
gateway inet (g)     : 170.11.52.2
user (u)             : rli
ftp password (pw) (blank = use rsh):
flags (f)            : 0x0
target name (tn)     : pxm45-71
startup script (s)   :
other (o)            :
```

burnboot

Burn Boot Software

Burns the specified *revision* of boot software on a standby AXSM card by specifying the *slot* number of the card and the *revision* number of boot software to burn.

Cards on Which This Command Runs

PXM45

Syntax

dspcd <*slot*> <*revision*>

Syntax Description

slot	The slot number of the standby AXSM card on which to burn the software.
revision	The revision number of the software to burn.

Related Commands

None

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

This example burns boot software version 2.0(1.166)D on slot 1.

```
MGX8850.7.PXM.a > burnboot 1 2.0(1.166)D
```

clidbxlevel

Command Line Interface Level—modify the contents of the CLI help output.

The **clidbxlevel** command lets you display the attributes for a command. You must execute **clidbxlevel** on each card where you want to change the level of displayed information. For example, if you execute **clidbxlevel** on an AXSM in slot 2 and want to see the same level of information in slot 8, you must execute **clidbxlevel** on the CLI in slot 8 and make the appropriate change. The attributes consist of:

- The required card state for execution
- The minimum user-privilege for the command
- Whether an incident of command execution appears in a log file

Cards on Which This Command Runs

PXM45

Syntax

clidbxlevel [*level*]

Syntax Description

level	The level is either 0–3. If you do not include a level, the system states the current level.

Related Commands

None

Attributes

Log: no log	State: active, standby, init	Privilege: SERVICE_GP

Example

Specify level 1 for the CLI. Obtain attributes for all “*user*” commands.

```
pop20one.7.PXM.a > clidbxlevel 1
Value of cliDbxLevel is now 1
pop20one.7.PXM.a > ? user

    Command          Access           Card         Log
    ---------------------------------------------------
    adduser          GROUP1           A             +
    cnfuser          GROUP1           A             -
    deluser          GROUP1           A             +
    dspusers         ANYUSER          A|S           -
    users            ANYUSER          A|S           -
```

clrallcnf

Clear All Configurations

Deletes the configuration of all the cards in the switch. After **clrallcnf**, you need to reconfigure the switch. (See **setrev**.)

The **clrallcnf** command clears all configuration information except for the boot IP address (see **bootchange**) and the time of day.

Caution Be absolutely sure you need to execute **clrallcnf** because it clears all configuration files on the PXM45. After you enter **clrallcnf**, the system prompts you to confirm the action.

Cards on Which This Command Runs

PXM45

Syntax

```
clrallcnf
```

Related Commands

None

Attributes

Log: no log	State: active, init	Privilege: SERVICE_GP

Example

Clear all the configuration elements for all the cards in the node. The system prompts for confirmation.

```
node1.7.PXM.a > clrallcnf
All SM's config will be deleted, and
        the shelf will be reset.
Do you want to proceed (Yes/No)?
```

clrcnf

Clear Configurations—clears a significant amount of the node configuration

Clears the configuration then reboots the switch. This command restarts the switch with a new configuration but keeps the basic configuration for the switch—IP connectivity, for example. The **clrcnf** command is useful if you frequently reset the switch but do not want to reconfigure basic settings. The information that is deleted and the retained information appear in the lists that follow.

The items that **clrcnf** clears are the:

- Connections
- Line and port configurations
- Resource partitioning
- Redundancy configuration for Y-cable or APS
- ATM address and PNNI configuration
- Other physical and logical provisioning

The following information is automatically saved then restored after reboot:

- IP address information for LAN, ATM, and SLIP
- SVC address information for ATM port for IP connectivity
- PVC address information for ATM port for IP connectivity
- Up to 51 records for user ID (login), passwords, and access levels
- CLI special configuration options (cntpAuthParams)
- One record for user-authentication parameters
- One record for long or short warnings for user-login
- Up to 25 records of user IDs
- RTMData (trap configuration for shelf IP and trap managers)
- Trap manager IP/port
- One record for shelf trap IP
- Correct primary and secondary software version for all slots
- SNMP community string, contact and location
- Node name
- Date, time, time zone, and GMT offset

Caution Be sure you need to execute **clrcnf** because it clears a significant number of configuration files. After you enter the command, the system prompts you to confirm the action.

Cards on Which This Command Runs

PXM45

Syntax

```
clrcnf
```

Syntax Description

This command takes no parameters.

Related Commands

None

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

Example

Clear all the configuration elements for all the cards in the node. The system prompts for confirmation.

```
node1.7.PXM.a > clrcnf
All SM's disk config will be deleted, and
        the shelf will be reset.
Do you want to proceed (Yes/No)?
```

cnfclkparms

Configure Clock Parameters—configure line characteristics for an E1 BITS clock

The **cnfclkparms** command lets you configure the *signal type* and *cable type* for E1 BITS sources. The configuration applies to both (upper and lower) sources.

Note In the current release, you can specify only the cable type.

Cards on Which This Command Runs

PXM45

Syntax

`cnfclkparms` `<signal type> <cable type>`

Syntax Description

signal type	Specifies whether the signal type for the clock is data or a synchronization signal. Enter a 1 for data or a 2 for synchronization.
cable type	Specifies whether the cable is a twisted pair or a coaxial cable. Enter a 1 for twisted pair or 2 for coaxial. The default is 1.

Related Commands

None

Attributes

Log: no log	State: active	Privilege: ANYUSER

cnfclksrc

Configure Clock Source—configure primary and secondary clocks and optional BITS clock

Configures a primary or secondary clock source for the node. A clock source can be:

- An external device that connects to the PXM-UI S3 card
- A line on an active AXSM

Clock Operation

When the switch first powers up, the internal oscillator on the PXM45 provides the clock to the node. Thereafter, you configure the clock sources at each node according to a well-designed plan for network synchronization. A typical configuration for an MGX 8850 network starts with a Building Integrated Timing System (BITS) clock source of stratum 3 or higher on one node. Therefore, the node with the BITS clock becomes the *master clock source* for the network. The active clock drives the clock line on the backplane, and each service module takes its clock from this line. Thereafter, the clock goes out through every line and is available as a configurable clock source on the other nodes.

Currently, automatic propagation of a master clock through the network is not available. To propagate the BITS-sourced clock to the other nodes, you execute **cnfclksrc** on the PXM45 at each node to specify primary and secondary clocks derived from the AXSMs.

(For a description of line-level *looped* timing, refer to the **cnfln** description the chapter, “Equipment and Resource Provisioning.” With looped timing, a clock arrives on a line and is redirected to become the transmit clock for only that line.)

Prerequisites to Clock Configuration

Whether it uses BITS or an AXSM line for a clock source, the node first must have a network controller. See the **addcontroller** description. For an AXSM-sourced clock, the additional prerequisites are:

- Activating the applicable line through **upln**
- Creating *logical* ports through **addport**
- Creating resource partitions through **addrscprtn**

Database Updates and Clock Configuration

If the node has a redundant PXM45, it automatically receives changes you make to the clock configuration as well as automated clock changes that occur under node management. For example, if you delete a clock source (**delclksrc)**, the standby card automatically implements this configuration change. Also, any switch from primary to secondary source is recorded by the standby PXM45.

Cards on Which This Command Runs

PXM45

Syntax

The syntax for **cnfclksrc** depends on the clock source.

For the external BITS clock:

`cnfclksrc` *`<priority> <portid>`*
portid has the format [*shelf*.]*slot.port* –**bits** e1 | t1 [–**revertive** <enable | disable>]

For AXSM-sourced clock (note the positions of the periods and colons):

`cnfclksrc` *`<priority> <portid>`*
portid has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]

Syntax Description

priority	The priority of the clock source is either *primary* or *secondary*. The default is primary.
portid for BITS	• *shelf* is always 1 and is purely optional. • *slot* is the logical slot number 7 for a BITS circuit on the PXM45-UI S3 (regardless of where the active PXM45 resides). • *port* is a logical number that indicates the upper or lower external clock connector on the PXM45-UI S3. The logical port number for the upper connector is 35. The lower connector is 36. • **bits**—a required keyword once you specify slot number 7 and a port number of 35 or 36 because you have identified a BITS clock source. Type the string "–bits" followed by a space then either "e1" or "t1." See "Usage Guidelines" for details. • **revertive**—an option that applies to only the BITS clock. Type the string "–**revertive**" followed by the complete word "enable" or "disable." The default is disable. See "Usage Guidelines for cnfclksrc" for important details.
portid for AXSM	• *shelf* is always 1 and is purely optional. • *slot* is the slot number of the AXSM. • *subslot* identifies the upper or lower bay of the back card—either a 1 for the upper bay or 2 for the lower bay (default is 1). • *port* is the line number on the AXSM. (The specified line must already be active (see **upln**). • *subport* is the logical port number in the range 1–60. This value is the *logical* port (or *ifNum*) that you must have assigned through **addport**.

Usage Guidelines for cnfclksrc

This section contains guidance for using **cnfclksrc** and important details about its parameters.

Specifying Primary and Secondary Clock Sources

Before using **cnfclksrc**, note the following:

- For a user-configured clock, the controller must have been specified by using **addcontoller**.

- AXSM-sourced clocks require that the lines, ports, and resource partitioning have been configured.
- A switch can have one primary source and one secondary source.
- For each execution of **cnfclksrc**, you can specify only one clock source (either but not both primary and secondary). Therefore, you must repeat **cnfclksrc** to specify the other clock source.
- If you do not to specify a secondary source, the internal oscillator serves as the secondary source.
- For clock sources on the AXSMs, Cisco recommends that primary and secondary sources be on separate cards or at least on separate lines.
- Revertive mode applies to only a primary BITS clock. For more details on the revertive option, see the section, "Configuring a BITS Clock."
- The switch constantly monitors the state of the clocks. For information on clock alarms, see the **dspclkalms** description.

Changing the Priority of a Clock Source

To change the priority of a clock source, the command sequence depends on the priority of the sources:

- To change the priority of a clock from primary to secondary or secondary to primary, you must first execute **delclksrc** to deconfigure each source.
- To change from one primary source to another primary source, you need to execute only **cnfclksrc** for the new primary source—the system automatically deconfigures the existing primary source.

Configuring a BITS Clock

You can configure a node to obtain its primary and secondary clocks through the BITS circuitry on the PXM-UI S3. (The PXM-UI S3 has two connectors to receive highly stable clocks from an external device. The PXM-UI S3 can support stratum levels 1–3.) If the primary and secondary clocks are externally-sourced, they must be the same rate. For example, you cannot specify a T1 primary source and an E1 secondary source.

Note Whenever the internal oscillator becomes the primary or secondary source due to a failure, a minor alarm is triggered on the local node.

You can enable a *revertive* mode for the primary BITS clock. The revertive function on the PXM45 applies when the primary clock source fails. A failure is a loss of the primary clock source after the node has locked to that clock source. If a primary clock recovers from a failure and revertive mode is enabled, the node automatically reverts to the primary source. The restored primary clock must be available for 12 seconds before it again becomes the active clock source.

If the primary clock source fails and revertive mode is disabled, you must re-configure the primary source after the failure has been corrected.

To change the mode from revertive to non-revertive, execute **cnfclksrc**. Follow the portID and priority with "**–revertive** disable."

Note For an E1 BITS clock, the current product is automatically limited to two parameters of an E1 line that is used as a BITS source: twisted pair cabling and date-type signaling.

Related Commands

dspclksrcs, **delclksrc**, **dspclkalms**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Configure the E1 clock at the upper connector of the PXM-UI S3 as the primary source. Configure subport (logical port) 10 on the line of the AXSM-1-2488 in slot 3 as the secondary. For the secondary source on the AXSM, note the locations of the periods and colons. Upon successful execution, the system displays a confirmation message.

```
pinnacle.7.PXM.a> cnfclksrc primary 7.35 -bits e1
Clock Manager has been successfully executed.

pinnacle.7.PXM.a> cnfclksrc secondary 3:1.1:10
Clock Manager has been successfully executed.
```

Configure a primary network clock to revert to the highest priority E1 clock source after recuperation from a failure. Upon successful execution, the system displays a confirmation message.

```
pinnacle.7.PXM.a> cnfclksrc primary 7.36 -bits e1 -revertive enable
Clock Manager has been successfully executed.
```

cnfdate

Configure Date

Configure the system date. The system does not return a message unless an error occurred. To see the date, execute **dspdate**.

Cards on Which This Command Runs

PXM45

Syntax

`cnfdate` *`<mm/dd/yyyy>`*

Syntax Description

mm/dd/yyyy	• *mm* is the month in the range 01–12. • *dd* is the day in the range 01–31. • *yyyy* is the year in the range 0000–9999.

Related Commands

dspdate

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Set date to June 26, 2000

```
excel.1.3.PXM.a > cnfdate 06/26/2000
```

cnfname

Configure Name—specifies a name for the node.

The case-sensitive node name must begin with a letter. It can include:

- Up to 32 letters or numbers
- Two special characters (“_” and “-”)
- No spaces

After you enter the name, the system prompts you for confirmation. To see the configured name, execute **dspcds** (or many of the other node-level display commands): the node name is the first item in the display.

Cards on Which This Command Runs

PXM45

Syntax

`cnfname`
`<node name>`

Syntax Description

node name	Node name consisting of up to 32 alpha-numeric characters.

Related Commands

None

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Configure the node name to be “MGX8850.” The system requests you to confirm the name. The CLI prompt returns with the new name. In this example, however, the name as it appears in the prompt is truncated to eight characters because of space limitations for information displayed in the prompt.

```
NODENAME.7.PXM.a > cnfname MGX8850
This node name will be changed to MGX8850. Please Confirm
Do you want to proceed (Yes/No)?
MGX8850.7.PXM.a >
```

cnfndparms

Configure Node Parameters—configure diverse parameters for the node.

The **cnfndparms** command lets you specify numerical values or enables for a diverse set of node-level parameters. This command has parameters that consist of an option number and a value or a yes/no option. The configuration resides in non-volatile RAM and thus survives a system reset or power cycle.

The paragraphs that follow describe each option as well as the ranges for each type of number. For information on the related alarms, see **dspndalms** and **dspenvalms**.

Maximum Card Resets

Options 1 and 2 let you specify a window of time in seconds and a number of resets to count the number of card resets. The purpose of these parameters is to prevent an endless loop of card resets.

- Option 1 lets you select the sliding window of time for counting the resets of the shelf management cards. The characteristics of the time period option are:
  - The units of measure are seconds.
  - The number is a 16-bit decimal number and therefore has the range 0–65355.
  - A 0 means an infinite time period. The impact of an infinite time period is that only a specified count of resets can stop the resets.
  - The default is 3600 seconds (1 hour).
- Option 2 lets you select the maximum number of resets of the shelf management card group per time period. Its characteristics are:
  - The number is an eight-bit decimal number and therefore has the range 0–255. The meaning of a 0 for this parameter is an infinite number of resets—the resets can continue indefinitely.
  - The default is 3 resets per time period.

Shutting Off Alarms for Absent Core Cards

Option 4 lets you specify whether an absent, redundant core card causes an alarm. (The core card the PXM45.) The purpose of this option is to let you turn off alarms when the node configuration shows core card redundancy but one card stays out of the backplane for an extended period of time. In essence, it lets you turn off alarms until you re-install the card.

Enable Expanded Memory on PXM45/B

Option 5 lets you enable expanded memory on the PXM45. To enable expanded memory, a pair of PXM45/Bs must reside in the system.

Required Power Supply Module Bitmap

Option 6 lets you specify the locations of *required* power supplies in an AC-powered system. If any one of the required supplies is removed, an alarm results. (See also the descriptions of **dspndalms** and **dspenvalms** regarding alarms.) Additional supplies can also exist in the power supply tray, but removing one of the additional supplies does not cause an alarm.

An AC power supply tray holds six power supply units (PSUs). (Refer to the *Cisco MGX 8850 Hardware Installation Guide* for details.) A supply belongs to one of two groups: A1–A3 or B1–B3. An 8-bit hexadecimal number identifies an individual supply. The value for Option 6 can be the sum of any combination of hexadecimal numbers. For example, the value for requiring A1 and B1 is:

0x01 + 0x10 = 0x11

Required Fan Trays

Option 7 lets you specify required fan trays for the purpose of alarm generation. You can specify either or both fan trays as required. The value is an 8-bit hexadecimal number.

Note The switch requires two fan trays for cooling purposes regardless of the number you specify for alarm purposes with **cnfndparms**.

- 0x0 means no fan try requirement. (The enclosure still must have at least one fan tray for cooling.
- 0x01 refers to the bottom fan tray.
- 0x02 refers to the top fan tray.

To require top and bottom fan trays, for example, enter a hexadecimal 3 for the option value:

0x01+0x02=0x03

Option Values

This scheme allows substantial flexibility for the numeric value of current and future options. This command supports various number formats as a part of its flexibility, as follows:

- An 8-bit decimal has the range 0–255.
- A 16-bit decimal number has the range 0–65535.
- A 32-bit decimal number has the range 0–4294962795.
- An 8-bit hexadecimal number has the range 0–0xff.
- A 16-bit hexadecimal number has the range 0–0xffff
- A 32-bit hexadecimal number has the range 0–0xffffffff.

Each option description states the type of number involved and the actual range for that option. Alternatively, the description states if the choice is “yes” to enable of “no” to disable.

Cards on Which This Command Runs

PXM45

Syntax

cnfndparms *<option_number>* *<option_value>*

Syntax Description

option number	A number that selects the option. The current parameters begin with number 2: • Option 1 is the number of seconds to count the resets of the shelf management cards. The range is 0–65355 (a 16-bit decimal number). The default is 3600 seconds (1 hour). A 0 means an infinite time period. The impact of an infinite time period is that only a specified count of resets can stop the resets. • Option 2 is the maximum number of resets of the shelf management card group per time period. The number is an 8-bit decimal number and therefore has the range 0–255. The meaning of a 0 for this parameter is an infinite number of resets—the resets can continue regardless of the number. The default is 3 resets per period (see Option 2). • Option 3 lets you enable or disable core card redundancy. Enter “yes” to enable or “no” to disable alarms on a missing, redundant core card. The default is enable, which means an alarm appears in the absence of a redundant core card. • Option 4 lets you enable or disable expanded memory on the PXM45/B. Enter “yes” to enable or “no” to disable The default is disable. • Option 5 you specify the locations of *required* power supplies in an AC-powered system. The number is 8-bit hexadecimal: – 0x0 (the default) means no specified power supply requirement related to this particular form of alarm generation (although the configuration must still meet the power requirements of the switch). – 0x01: PSU A1 is required. – 0x02: PSU A2 is required. – 0x04: PSU A3 is required. – 0x10: PSU B1 is required. – 0x20: PSU B2 is required. – 0x40: PSU B3 is required. • Option 6 lets you specify the location of one or more required fan trays. The number is 8-bit hexadecimal: – 0 for no specific fan try requirement – 0x01 for bottom fan tray required – 0x02 for top fan tray required
option value	The *option value* can be a decimal or hexadecimal number or a “yes” or “no” entry. The following shows the possible ranges or values for each type of numeric option. 8-bit decimal: 0–255 16-bit decimal: 0–65535 32-bit decimal: 0–4294962795 8-bit hexadecimal: 0–0xff 16-bit hexadecimal: 0–0xffff 32-bit hexadecimal: 0–0xffffffff

Related Commands

dspndparms, **dspndalms**, **dspenvalms**

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

Change the time period for counting card resets to 30 minutes. First enter the command with no parameters. The system displays the options and prompts you to enter an option number (rather than display the help information for this command). Enter a 2, then the system displays the current value for this option.

```
MGX8850.7.PXM.a > cnfndparms
MGX8850                         System Rev: 02.01   Sep. 04, 2001 12:15:13 PST
MGX8850                                              Node Alarm: CRITICAL
NODE CONFIGURATION OPTIONS
Opt#  Value       Type           Description
----  -----       ----           -----------
1     3600        16bit Decimal  SHM Card Reset Sliding Window (secs)
2     3           8bit Decimal   SHM Max Card Resets Per Window (0 = infinite)
3     Yes         Boolean        Core Redundancy Enabled
4     No          Boolean        Expanded Memory on PXM45B Enabled
5     0x0         8bit Hex       Required Power Supply Module Bitmap
6     0x0         8bit Hex       Required Fan Tray Unit Bitmap

Enter option number (1-5): 1
NODE CONFIGURATION OPTIONS
Opt#  Value       Type           Description
----  -----       ----           -----------
1     3600        16bit Decimal  SHM Card Reset Sliding Window (secs)
Interval used to measure maximum number of card resets allowed.
    Value must be greater than zero.
    If nonzero, number of seconds used to measure card resets.
```

Enter 1800 at the prompt. The system subsequently displays the new setting.

```
Enter value for option 1: 1800
NODE CONFIGURATION OPTIONS
Opt#  Value       Type           Description
----  -----       ----           -----------
1     1800        16bit Decimal  SHM Card Reset Sliding Window (secs)
```

You can enter the option number and option value without prompting, and the system shows the result.

```
MGX8850.7.PXM.a > cnfndparms 1 1800
NODE CONFIGURATION OPTIONS
Opt#  Value       Type           Description
----  -----       ----           -----------
1     1800        16bit Decimal  SHM Card Reset Sliding Window (secs)
```

cnfpasswd

Configure Password

Change your own password. After you enter the **cnfpasswd** command without parameters, the system prompts you to enter the new password then prompts you to re-enter it.

Note The default password is for a user-account is *newuser*.

Cards on Which This Command Runs

PXM45

Syntax

cnfpasswd
<password>

Syntax Description

password	Your new password.

Related Commands

adduser, **dspusers**, **cnfuser**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Change your password. After you enter the command, it prompts you once to enter a new password then prompts you to re-enter it.

```
pinnacle.8.PXM.a > cnfpasswd
Enter password:
Re-enter password:
```

cnfserialif

Configure Serial Interface

The **cnfserialif** command lets you change the data rate on a serial interface on the PXM45-UI-S3 back card. The two types of serial ports are the console port and the maintenance port. These ports provide user-access for controlling the switch. The default speed n a serial interface is 9600 bits per second, but higher speed terminals are frequently available.

Each port connects to a different type of terminal implementation. Refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide* for a description of how to use these physical ports for switch control.

Cards on Which This Command Executes

PXM45

Syntax

`cnfserialif` *`<port#>`* *`<speed>`*

Syntax Description

port#	Specifies the physical port: • 1=maintenance port. • 2=console port.
speed	Specifies a data rate in bits per second. Valid entries are 1200, 2400, 4800, 9600, 19200, and 38400.

Related Commands

delserialif, **dspserialif**

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

Configure the console port to have a data rate of 19200 bits per second.

```
node19.8.PXM.a > cnfserialif 2 19200
```

cnfsnmp

Configure SNMP Strings

Configure the SNMP strings. The three strings are *community*, *contact*, and *system location*. You can configure only one of these strings with a single execution of **cnfsnmp**.

Cards on Which This Command Executes

PXM45

Syntax

```
cnfsnmp
-community [string <ro | rw>]
-contact [string]
-location [string]
```

Syntax Description

-community	Keyword that establishes the community access string to permit access to SNMPv1 protocol. The *string* acts like a password and permits access to the SNMP Protocol. Further, the access of either read-only or read-write allows operations on MIB Objects according to the setting. The setting can be either “ro” for read-only or “rw” for read-write. The default is read-only With read-only, authorized management stations are only able to retrieve MIB objects. With read-write access, authorized management stations are able to retrieve and modify MIB objects.
-contact	Keyword that specifies the system contact string for sysContact MIB object in MIB-II. The string in this case is text that describes the contact. For example, the contact could be an administrator’s email address. The default is no text.
-location	Keyword that specifies the location of the system. The default is no text. The system location string is used for **sysLocation** MIB object in MIB-II.

Attributes

Log: log	State: active	Privilege: SUPER_GP

Related Commands

dspsnmp

Example

Configure various community strings.

```
node19.8.PXM.a > cnfsnmp community ro
node19.8.PXM.a >cnfsnmp community comaccess
node19.8.PXM.a >community string “comaccess”, read-only access
node19.8.PXM.a >cnfsnmp community comaccess ro
node19.8.PXM.a >community string “comaccess” read-only access
node19.8.PXM.a >cnfsnmp community superaccess rw
node19.8.PXM.a >community string “superaccess”, read-write access
```

Give an E-mail address for t he system contact

```
node19.8.PXM.a > cnfsnmp contact Dial System, Email:
```

Specify the location of the system as Building 3, Room 214.

```
node19.8.PXM.a >cnfsnmp location Building 3/Room 214
```

cnftime

Configure Time

Configures the time for the node. To see the time after you execute **cnftime**, use **dspdate**. The system displays the time in 24-hour format.

Note Configure a time zone through **cnftmzn** and optional GMT offset through **cnftmzngmt** before you configure the time through **cnftime**.

Cards on Which This Command Runs

PXM45

Syntax

`cnftime` `<hh:mm:ss>`

Syntax Description

hh:mm:ss	The format for time specification is: • *hh* is the hour in the range 01–24. • *mm* is the minute in the range 01–60. • *ss* is the second in the range 01–60.

Related Commands

cnfdate, **cnftmzn**, **dspdate**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Set time for 2 PM. plus 11 minutes and 22 seconds.

```
excel.1.3.PXM.a > cnftime 14:11:22
```

cnftmzn

Configure Time Zone

Configures the time zone in the Western Hemisphere for the switch. To configure a time zone outside the four standard time zones of the Western Hemisphere, enter the GMT argument, then execute **cnftmzngmt** to specify an offset in hours from Greenwich Mean Time.

The system returns no messages unless an error occurs. To see the time zone, execute **dspdate**.

Cards on Which This Command Runs

PXM45

Syntax

`cnftmzn` <*timezone*>

Syntax Description

timezone	The possible time zones requires all uppercase characters. GMT, Greenwich Mean Time EST, Eastern Standard Time CST, Central Standard Time MST, Mountain Standard Time PST, Pacific Standard Time EDT, Eastern Daylight Time CDT, Central Daylight Time MDT, Mountain Daylight Time PDT, Pacific Daylight Time

Related Commands

cnftime, **cnfdate**, **cnftmzngmt**, **dspdate**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Configure the time zone in the node to U.S. Pacific Standard Time.

```
excel.1.3.PXM.a > cnftmzn PST
```

cnftmzngmt

Configure Time Zone Relative to GMT

Configures the time zone for the node relative to GMT. Typically, this command applies to nodes outside the four standard time zones of the Western Hemisphere. Use **cnftmzngmt** according to the following sequence:

- First use **cnftmzn** to specify the time zone as GMT.
- Then specify an offset in hours relative to Greenwich Mean Time by executing **cnftmzngmt**. The values are GMT plus or minus an integer in the range 1–12.

Use **dspdate** to see the time.

Cards on Which This Command Runs

PXM45

Syntax

`cnftmzngmt` `<`*`timeoffsetGMT`*`>`

Syntax Description

timeoffsetGMT	Number of hours offset from GMT in the range -12 through 12.

Related Commands

cnftmzn, **cnftime**, **cnfdate**, **dspdate**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Set time zone in the shelf to GMT plus 4 hours.

```
excel.1.3.PXM.a > cnftmzngmt 4
```

cnfuser

Configure User

Configure a new password or privilege level for a user. If the user does not already exist, executing **cnfuser** with a new user-name creates that user.

If you do not specify a user-name (*userID)* but include one or more of the other parameters, the command applies to the current user.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfuser
-u <userID>
[-p <password>]
[-l <accessLevel>]
[-i <expiration interval>]
```

Syntax Description

-u	Keyword that specifies a string of 1–12 characters that identifies a user. The maximum number of users a system can accept is 50.
-p	(Optional) Keyword that specifies a new password with 5–15 characters for *userId*.
-l	(Optional) Keyword that specifies a new access level for the user. The *accessLevel* can be SERVICE_GP, SUPER_GP, GROUP1, or ANYUSER. The new level you type must be lower than the privilege of the current user. See **adduser** description for an explanation of privilege levels.
-i	(Optional) Keyword that specifies the password expiration interval. The password expiration interval is the number of days that a password is valid. At the end of those days, the user must change the password. The range for the expiration interval is 1–1000 days. (See **cnfpswdexpire** and **dsppswdexpire**).

Related Commands

adduser, **deluser**, **dspusers, cnfpswdexpire**, **dsppswdexpire**

Attributes

Log: no log	State: active	Privilege: GROUP1

Example

Change the password and privilege lever of user “rocky.” New password is “nevermind,” and the privilege level is GROUP1. Note that the you must be logged in at a higher than GROUP1 privilege level to specify GROUP1 for “rocky.” If the “-u” and userID (rocky) were not entered, this command would change the password and privilege of the current user.

```
raviraj.7.PXM.a > cnfuser -u rocky -p nevermind -l GROUP1 -i 20
```

Example screens of a user changing the password when password expiration check is On:

```
MGX8850.8.PXM.a >

Login: superuser
password:

Password has expired.
New password:
Re-enter new password:
ERR: Password too short (minimum: 5 characters)
New password:
Re-enter new password:
ERR: Twice-entered passwords mis-match
New password:
Re-enter new password:
ERR: Password too short (minimum: 5 characters)
You have exceeded the number of attempts allowed.

MGX8850.8.PXM.a >

Login: superuser
password:

Password has expired.
New password:
New password:
New password:
You have exceeded the number of attempts allowed.
```

commitrev

Commit Revision

Completes a graceful upgrade by committing to the operating firmware image as the primary version. The **commitrev** command is the necessary conclusion to a graceful upgrade. See the **loadrev** description for more details about graceful firmware changes.

The impact of **commitrev** is:

- It signifies that the primary firmware image activated through the **runrev** command is accepted.
- The previous image is removed from the card's main memory (but continues to reside on disk).
- Starting another graceful revision change becomes possible. If you attempt **loadrev** on the same card before you execute **commitrev**, the system blocks **loadrev** and states that a revision change is in progress.
- You cannot use **abortrev** to revert to the previous image. To bring a previous image into memory and run it, you must use **setrev** to force-load the image (a non-graceful revision change) or execute **restoreallcnf**.

The order of commands in a graceful upgrade, including the option of aborting the revision change, appears in the following list. For clarification of where firmware resides after each stage of the upgrade, refer to Table 2-7 for a single card and Table 2-8 for a redundant card pair.

1. **loadrev** loads a firmware version from the hard disk to a card's memory. In a non-redundant card setup, **loadrev** does not cause the system to reset the card.
2. **runrev** causes the primary card to start running the new version. For a redundant pair of cards, the standby becomes the active card then starts running the new version.
3. If an unacceptable problem occurs, the optional **abortrev** command restores the previous version of firmware as well as the previous database contents.
4. **commitrev** declares the new primary version to be acceptable and removes the old primary from main memory (but not the hard disk).

The stages of a graceful upgrade and the reset actions appear in Table 2-7 and Table 2-8. For a single-card upgrade, see Table 2-7. For a redundant-pair upgrade, see Table 2-8. The tables start by showing that, initially, the primary and secondary versions of firmware are 2.x, so the only possible operational version is 2.x. The **loadrev** command loads a generic version called 2.y, and the upgrade sequence progressively changes the primary and secondary firmware versions. If you execute **abortrev** before **commitrev**, one or two cards (redundant pair only) are reset, as the tables show.

Table 2-7 Single-Card Upgrade From 2.x to 2.y

Firmware Status	Initial Version	After loadrev	After runrev	After commitrev
Primary	2.x	2.x	2.y	2.y
Secondary	2.x	2.y	2.x	2.y
Current	2.x	2.x	2.y	2.y
			After **abortrev**, the card is reset.	

Note Of special note in Table 2-8, **runrev** causes the standby card to become the active card and run the new version of firmware. The reversed location of the "Active" and "Standby" columns shows these changed states.

Table 2-8 Redundant Pair Upgrade From 2.x to 2.y

Firmware status	Before upgrade		After loadrev		After runrev		After commitrev	
	Active	Standby	Active	Standby	Standby	Active	Standby	Active
Primary	2.x	2.x	2.x	2.x	2.y	2.y	2.y	2.y
Secondary	2.x	2.x	2.y	2.y	2.x	2.x	2.y	2.y
Current	2.x	2.x	2.x	2.y	2.y	2.y	2.y	2.y
			abortrev resets the standby card.		**abortrev** resets both cards.			

Cards on Which This Command Runs

PXM45

Syntax

commitrev *<slot>* *<revision>*

Syntax Description

slot	Number of the slot where firmware must revert to previous version.
revision	Revision number derived from the name of the firmware file. For an explanation, see the section, "Version Numbering Conventions," in the **loadrev** description.

Related Commands

loadrev, **abortrev**, **runrev**, **setrev**, **dspversion**

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

Commit the PXM45 in slot 8 to the graceful upgrade to file pxm45_002.000.000.000_mgx.fw (so 2.0(0) is the *version*). The system prompts you to confirm that you want the command to execute.

```
pinnacle.8.PXM.a > commitrev 8 2.0(0)
```

delclksrc

Delete Clock Source

Deletes a user-specified primary or secondary clock source. Changing a clock source or changing the priority of the source (primary or secondary) are the most frequent uses of **delclksrc**. See the description of **cnfclksrc** for these common uses of **delclksrc**.

Note If the node has a redundant PXM45, it automatically receives changes you make to the clock configuration as well as automated changes to clock status that occur under node management. For example, executing **delclksrc** is a configuration change that the standby card automatically implements. Also, a switch from primary to secondary clock source is also recorded by the standby PXM45.

Cards on Which This Command Runs

PXM45

Syntax

```
delclksrc <priority>
```

Related Commands

cnfclksrc, **dspclksrcs**, **dspclkalms**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Delete the primary clock source.

```
pinnacle.7.PXM.a> delclksrc primary
```

deltrapmgr

Delete Trap Manager

Delete a trap manager. The **deltrapmgr** command requires an IP address for deletion. To see existing trap managers, use **dsptrapmgr**. For more information about trap managers, see the *Cisco MGX 8850 Routing Switch Software Configuration Guide* or the **addtrapmgr** description in this book.

Cards on Which This Command Runs

PXM45

Syntax

deltrapmgr <*ip_addr*>

Syntax Description

ip_addr	IP address in dotted decimal format: *nnn.nnn.nnn.nnn*, *n*=0-9 and *nnn* < 256

Related Commands

addtrapmgr, **dsptrapmgr**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Delete trap manager with IP address 161.10.144.56.

```
node501.7.PXM.a > deltrapmgr 161.10.144.56
```

deluser

Delete User

Removes a user from the list of users on an MGX 8850 node. The system does not allow you to delete a user with a privilege level higher than the level at which you execute the command. For example, if the current user privilege is 2 (GROUP2), you cannot delete a user at level 1 (GROUP1). See the **adduser** description for the user-privilege hierarchy. No screen output appears unless an error occurs.

Cards on Which This Command Runs

PXM45

Syntax

```
deluser
<user ID>
```

Syntax Description

user ID	User name, consisting of up to 12 characters.

Related Commands

dspusers, **adduser**

Attributes

Log: log	State: active	Privilege: GROUP1

downloadflash

Download Flash—load the first boot code found by the PXM45 hard drive into flash memory.

The **downloadflash** command does not execute at the runtime prompt. It operates in bootmode only.

A **downloadflash** session concludes the sequence of tasks for performing a PXM45 boot code load. Prior to executing this command, you must access the boot code and transfer the file to the PXM45 hard drive by using a “put” command). Arguments within the “put” command let you load boot code to any combination of standby or active PXM45s. (See Example section for details.) Once firmware is installed in slot 7, the bootcode is mirrored to a new PXM45 in slot 8 if present. However, to ensure that the boot code is correct, use **downloadflash** as a manual way to download the boot code to the standby PXM45.

Note Make sure only one version of backup boot code resides in the firmware directory: either delete or rename old versions to ensure that **downloadflash** uses the correct version.

Cards on Which This Command Runs

PXM45

Syntax

downloadflash

Related Commands

None

Attributes

Log: no log	State: active, standby	Privilege: SUPER_GP

Example

Do a PXM45 boot code load. Start with a tftp to the boot code source. Conclude with the download to the standby and the active PXM45. Despite the “.fw” argument in the command string, this is NOT a firmware load. The first lines show an attempt to run downloadflash within the runtime image.

```
Unknown.7.PXM.a > downloadflash
Error: flash_file supported only at backup boot

> ftp <switch_dest_addr>
> bin
> put <pxm_bkup_version>.fw PINNACLE@PXM45.BT
> quit
wilco.7.PXM.a > downloadflash
```

- To place the boot code on the active PXM45 only, use the following “put” string:
  >put pxm_bkup_version>.fw PINNACLE@PXM45_ACTIVE.BT
- To place the boot code on the standby PXM45 only, use the following “put” string:
  >put pxm_bkup_version>.fw PINNACLE@PXM45_STANDBY.BT

dspbkpl

Display Backplane—display details about the backplane.

The **dspbkpl** command shows the following types of information about the backplane:

- Card type (a hexadecimal number)
- Chassis-level part number and revision number
- PCB 73-level part number
- Chassis serial number
- CLEI code
- PCB 28-level part number

Cards on Which This Command Runs

PXM45

Syntax

dspbkpl

Syntax Description

This command takes no parameters.

Related Commands

None

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display details about the backplane.

```
Unknown.8.PXM.a > dspbkpl

BackPlane Information
---------------------

Card Type:                      0x12b
Chassis 800-level P/N:          800-03145-07
Chassis 800-level rev:          B0
PCB 73-level P/N:               73-3094-02
Chassis serial number:          SAA03190530
CLEI CODE:                      IPMAABOARA
PCB 28-level P/N:               28-2681-02
```

dspcd

Display Card

Display the following information about a card:

- Hardware serial number.
- Firmware revision level. (See the **loadrev** description for an explanation of how to interpret the revision field.)
- Status, possibly including the reason for the last reset (FunctionModuleResetReason) and state of the integrated alarm (cardIntegratedAlarm).
- For a service module only: a count of configured lines, ports, and connections.

Note The connection count includes control VCs when you execute **dspcd** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

- For a service module only: which physical lines constitute a port group and the maximum number of connections in that port group. A port group consists of one to many physical lines. This maximum connection count is a function of the hardware interface type (OC-3, OC-12, and so on). The port group information also shows the number of existing SVCs, SPVCs, and SPVPs.

  Use the maximum number of supported connections to help you configure resource partitions. If a particular resource partition has close to the maximum supported by hardware on a line, few or no connections would be possible in another partition on the same line.

Some of the information that **dspcd** shows can also be displayed using the **version** command, but **version** shows the boot code version in bold.

The total number of connections in the display includes control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that do not include control VCs, use **dsppnport**.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
dspcd
```

Syntax Description

This command does not take parameters.

Related Commands

dspcds, **version**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Examples

Display card details for the current PXM45.

Note The “A1” at the end of the primary software revision and boot firmware revision numbers shows that these versions are pre-release. Refer to the **setrev** description for details.

```
MGX8850.7.PXM.a > dspcd
MGX8850                         System Rev: 02.00   Aug. 02, 2000 23:39:06 GMT
MGX8850                                               Node Alarm: CRITICAL
Slot Number    7    Redundant Slot:  8

                    Front Card          Upper Card          Lower Card
                    ----------          ----------          ----------

Inserted Card:      PXM45               UI Stratum3         PXM HardDiskDrive
Reserved Card:      PXM45               UI Stratum3         PXM HardDiskDrive
State:              Active              Active              Active
Serial Number:      SAK033600AN         SBK044200J8         SAK0403005Q
Prim SW Rev:        2.0(14.8)P1         ---                 ---
Sec SW Rev:         2.0(14.8)P1         ---                 ---
Cur SW Rev:         2.0(14.8)P1         ---                 ---
Boot FW Rev:        2.0(14.8)P1         ---                 ---
800-level Rev:      12                  A0                  06
800-level Part#:    800-05983-01        800-05787-02        800-05052-03
CLEI Code:          0000000000          BA7IBCLAAA          0000000000
Reset Reason:       On Reset From Shell
Card Alarm:         NONE
Failed Reason:      None
Miscellaneous Information:
M8850_LA                         System Rev: 02.01   Sep. 05, 2001 17:18:12 PST
MGX8850                                               Node Alarm: CRITICAL

Crossbar Slot Status:      Present

Alarm Causes
------------
    NO ALARMS
```

Display card details for the current AXSM-16-155.

```
MGX8850.3.AXSM.a > dspcd
                    Front Card          Upper Card          Lower Card
                    ----------          ----------        ------------

Card Type:          AXSM-16-155         MMF-8-155-MT        ---

State:              Active              Present             Absent

Serial Number:      SAK0350008L         SAK0403004A         ---

Boot FW Rev:        2.0(11)A1           ---                 ---

SW Rev:             2.0(1)D             ---                 ---
```

```
800-level Rev:      M6                  07                  ---

Orderable Part#:    800-5776-3          800-4819-1          ---

PCA Part#:          73-4504-2           73-3845-1           ---

Reset Reason:On Power up

Card SCT Id: 2

Type <CR> to continue, Q<CR> to stop:
#Lines #Ports #Partitions   #SPVC   #SPVP    #SVC
------ ------ ----------- ------- ------- -------
     2      2           2      10       0      10

Port Group[1]:
#Chans supported:32512  Lines:1.1 1.2 1.3 1.4
Port Group[2]:
#Chans supported:32512  Lines:1.5 1.6 1.7 1.8
Port Group[3]:
#Chans supported:32512  Lines:2.1 2.2 2.3 2.4
Port Group[4]:
#Chans supported:32512  Lines:2.5 2.6 2.7 2.8
```

dspcds

Display Cards

Displays high-level information for all the cards in the node. For more detailed information about a card, execute **dspcd** on the CLI of that card. The information that **dspcds** provides is the:

- Revision level of the boot firmware
- Revision level of the system software
- Date and time of command execution, including GMT offset
- Backplane serial number and its hardware revision level
- The IP address of the statistics master (a workstation)
- Type of card in the front and back slots and the (active/standby) state of each
- Alarm status for each card and the shelf itself
- Redundancy configuration for each slot

Cards on Which This Command Runs

PXM45

Syntax

dspcds

Related Commands

dspcd, **version**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display information for all cards in the MGX 8850 switch.

```
Unknown.7.PXM.a > dspcds
Unknown                          System Rev: 02.00   Aug. 06, 2000 18:03:35 GMT
Backplane Serial No: SAA03270618 Bp HW Rev:  B0      GMT Offset: 0
                                                     Node Alarm: CRITICAL
Card  Front/Back       Card           Alarm      Redundant   Redundancy
Slot  Card State       Type           Status     Slot        Type
---   ----------       --------       --------   -------     -----

01    Active/Active    AXSM_1OC48     NONE       NA          NO REDUNDANCY
02    Active/Active    AXSM_1OC48     NONE       NA          NO REDUNDANCY
03    Active/Active    AXSM_16OC3     NONE       04          PRIMARY SLOT
04    Standby/Active   AXSM_16OC3     NONE       03          SECONDARY SLOT
05    Active/Active    AXSM_4OC12     NONE       NA          NO REDUNDANCY
06    Active/Active    AXSM_4OC12     NONE       NA          NO REDUNDANCY
07    Active/Active    PXM45          NONE       08          PRIMARY SLOT
08    Standby/Active   PXM45          NONE       07          SECONDARY SLOT
09    Active/Active    AXSM_16T3E3    NONE       NA          NO REDUNDANCY
10    Active/Active    AXSM_16T3E3    NONE       NA          NO REDUNDANCY
11    Empty            ---            ---        ---         ---
12    Active/Active    AXSM_2OC12     NONE       NA          NO REDUNDANCY
13    Empty            ---            ---        ---         ---
14    Empty            ---            ---        ---         ---
```

dspclksrcs

Display Clock Sources

Displays the configuration and status of the clock sources on the node. (For details about network synchronization, see the description of **cnfclksrc**.) The **dspclksrcs** output consists of:

- For the primary clock: the type, source, status, and reason (for status change) of the clock.
- For the secondary clock: the type, source, status, and reason (for status change) of the clock.
- The active clock—the clock that currently provides synchronization. The active clock can be primary, secondary, holdover, or internal.
- Whether revertive mode is enabled or disabled.

Note Changes to the configuration and status of clocks go into the database on the active PXM45. If a standby (redundant) PXM45 exists, it receives the initial clock configuration and status but receives internal status updates only when you interact with the node in a way that changes a configuration or when the standby PXM45 switches to the active state.

Type of Clock Source

The type is either BITS or generic. Currently, generic applies to only an AXSM-sourced clock. If a user-specified priority of clock is not configured, the source is *null*. For the current release, the null source is presumed to be the internal oscillator.

Possible Sources

The *source* of the clock has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. More typically, the source has the two-part, short-hand form *slot.line* or *slot.port*. If the source is an AXSM, the format is *slot.line*.

For a BITS clock, the format is *slot.port*. The slot for a BITS clock is 7. The logical port is always 35 or 36. Port 35 refers to the upper external clock connector, and port 36 refers to the lower connector.

Clock Status

The status of a particular clock source can be one of the following:

- “ok” (good), which means the clock source is operational and stable.

  (If the *status* is “ok,” then the *Reason* field shows “okay.” If the status is “ok,” the reason for the status change described in the section, “Reason for Status Change,” is not important.)
- “bad” means a fault in the clock source has been detected. Use the Reason field to help isolate the problem. See the section, “Reason for Status Change.”
- “unknown” is a temporary string while the clock manager is sending a message to the clock source.
- “not configured” means that this source—primary or secondary—has not been configured.

Reason for Status Change

The reasons that clock status can change are numerous. The **dspclksrcs** command displays a Reason field for both the primary and the secondary clock source. The reason can include the first-time, user-specification of the clock source. The reason strings and their meaning appear in Table 2-9. Additional information about “okay” and the locking process follows.

Table 2-9 Reasons for Change of Clock State

Reason	Meaning
okay	The clock source is okay.
unknown reason	The clock manager has no information for Reason.
no clock signal	Loss of signal (LOS) on the clock source.
frequency too high	The frequency has drifted too high.
frequency too low	The frequency has drifted too low
excessive jitter	Jitter has exceeded tolerance for this stratum.
missing card or component	The active PXM45 has no clock hardware support.
non-existent logical interface	The interface is non-existent or not functioning.
interface does not support clocking	The interface does not support clocking.
phase error	The clock manger has detected a phase error in the clock.
unlockable	The clock manager has attempted to lock the source but found that the clock signal from this source is unlockable.
out of lock or null	The clock circuitry is again trying to lock a source that has gone out of locking range. Note that for Reason, "out of lock" and "null" is synonymous.
reset—not a valid state	The clock source has been reset.
in locking—wideband test	The clock circuitry is in wide bandwidth mode of the locking process. In this mode, the circuit tests the integrity of the source but with wide latitude for frequency accuracy. If the source passes this test, the circuit proceeds to the narrowband test.
in locking—narrowband test	The clock circuitry is in narrow bandwidth locking mode. In this mode, the circuit stringently tests the integrity of the source.
locked	The clock circuitry is locked to this source.

When you configure a new clock source or the current clock source changes due to any reason, the software goes through the process of validating the new, *current* clock source again. (For example, the reasons other than direct user-configuration can be: the previous clock source goes out of lock or a re-synchronization of the clock sources takes place due to a switch-over or a rebuild.) This validation process takes the current clock source through the following states:

- in locking—wideband test
- in locking—narrowband test
- locked

During these states, the node is already using the new clock source as the synchronizing source.

You might also see these states—in the sequence previously listed—if the current clock source was momentarily lost because it drifted out of the lockable range for either the frequency or the phase. In such a case, the software goes through one more round of trying to confirm that the current clock source is lockable before it declares a clock source to be unlockable. If the software finds that, even after this repeated attempt, that the clock source is not coming back within the lockable range, it declares the clock

source as unlockable and proceeds to use the next clock in the hierarchy (of primary, secondary, internal oscillator) as the current clock source. The exception to this final validation scenario occurs if the current clock source is the internal oscillator in either the *free running* mode or the *hold-over* mode: in this case, the software omits this final validation attempt because no other clocks sources within a clock hierarchy are available.

Revertive Behavior

For information on revertive behavior, see the **cnfclksrc** description.

Cards on Which This Command Runs

PXM45

Syntax

dspclksrcs

Related Commands

cnfclksrc, **delclksrc**, **dspclkalms**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current clock sources. The display shows that both the primary and secondary clocks are good. They are sourced at lines 2 and 3 of the AXSM in slot 6. Also, the active clock is provided by the the primary source. The primary and secondary clock reason is “okay” in each case.

```
pinnacle.7.PXM.a> dspclksrcs
Primary clock type:     generic
Primary clock source:   6.2
Primary clock status: good
Primary clock reason:   okay
Secondary clock type:   generic
Secondary clock source: 6.3
Secondary clock status:   good
Secondary clock reason:   okay
Active clock:           primary
source switchover mode: non-revertive
```

Display information on the clock sources. The display shows that nothing has been configured, so the internal oscillator generates the primary and secondary clocks. The primary and secondary clock reason is “okay” in each case.

```
Unknown.7.PXM.a > dspclksrcs
Primary clock type:     null
Primary clock source:   0.0
Primary clock status:   not configured
Primary clock reason:   okay
Secondary clock type:   null
Secondary clock source: 0.0
Secondary clock status:   not configured
Secondary clock reason:   okay
Active clock:           internal clock
source switchover mode: non-revertive
```

Display information about the clock sources. This example shows a BITS clock for the primary source with revertive mode enabled.

```
pop20one.7.PXM.a > dspclksrcs
Primary clock type:     bits t1
Primary clock source:   7.35
Primary clock status:   ok
Primary clock reason:   okay
Secondary clock type:   generic
Secondary clock source: 9:1.1:1
Secondary clock status:   ok
Secondary clock reason:   okay
Active clock:           primary
source switchover mode: revertive
```

dspdisk

Display Disk

Display utilization for all partitions on the hard disk. The display shows the allocated space and the free space. A likely application of **dspdisk** is a routine check of disk utilization by running a script that includes this command.

Note The capacity of the disk is very large relative to typical usage and therefore does not present potential restrictions. The output shows the allocated space rather than the physical capacity of the drive.

Cards on Which This Command Runs

PXM45

Syntax

`dspdisk`

Syntax Description

This command takes no parameters.

Related Commands

cd

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the utilization for the default partition C.

```
orpswp3.2.PXM.a > dspdisk
==========================================================
 Partition C: Allocated Size:  800 MB Free Space:  574 MB
 Partition D: Allocated Size:  600 MB Free Space:  564 MB
 Partition E: Allocated Size:  100 MB Free Space:   99 MB
 Partition F: Allocated Size:  800 MB Free Space:  799 MB
==========================================================
```

dspipconntask

Display IP Connectivity Task

Display the current state of the IP connectivity task. As a part of a troubleshooting regimen, the **dspipconntask** command can help you isolate a problem related to IP connectivity.

Cards on Which This Command Runs

AXSM

Syntax

`dspipconntask`

Related Commands

ipifconfig, dspipif, **dspipifcache**, **setipconndebug**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the task information IP connection on the PXM45. Note that the Task Debug Level can be modified through the **setipconndebug** command.

```
MGX8850.7.PXM.a > dspipconntask

IP CONNECTIVITY TASK INFORMATION
----------------------------------------------------------
Task State:                  ACTIVE
Card State:                  READY
Task Id:                     0x10010
Subtask Id:                  0x1005c
Disk API State:              OK
SyncRam API State:           OK
Task SyncRam State:          UPDATE
Task Disk Update Bitmap:
    Device Table:     0  0  0
Task SyncRam Update Bitmap:
    Disk:             0  0  0
    IO Links:         0  0  0
    Interface Cache:  0  0  0
    BootChange Sync:  0  0  0
Task Debug Level:            0x1
Task Logging To:             Event Log
M8850_LA.7.PXM.a >
```

dspipif

Display IP Interface Configuration

Display configuration and other information for either one or all IP interfaces on the current PXM45. If you request all interfaces by entering **dspipif** with no parameters, the display shows information for all interface types. The displayed information comes from the current state of the interface and the configuration specified through **ipifconfig**. The information consists of the:

- Configuration (see **ipifconfig** for descriptions).
- Operational state.
- Statistics (including transmitted and received packets and errors).

Cards on Which This Command Runs

PXM45

Syntax

`dspipif`
`[`*`interface`*`]`

Syntax Description

interface	(Optional) An alphanumeric string that identifies a type of interface for display.Without this parameter, the system displays the configuration state of all interface types. The choices for *interface* are as follows: • **lnPci0** specifies the Ethernet interface (the default on power-up). • **atm0** specifies the ATM interface. • **sl0** specifies the SLIP interface

Related Commands

ipifconfig, **dspipifcache**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display information for all IP interfaces. The output shows that no configuration exists for the ATM interface but do for Ethernet and SLIP. Note that for each interface in the current release, the “unit number” has no meaning. The Flags field for Ethernet shows that the interface is UP, a broadcast address has been configured, ARP is enabled, and that the interface is running. (See the ipifconfig description for the meaning of these parameters. The output also shows the number of packets that have crossed the Ethernet interface. Although a configuration exists for SLIP, the display shows that no packets have crossed this interface.

```
pinnacle.7.PXM.a > dspipif
Unknown                          System Rev: 00.00   Jan. 04, 2000 12:16:22 GMT
MGX8850                                              Shelf Alarm: NONE
IP INTERFACE CONFIGURATION
--------------------------------------------------------------------
atm (unit number 0):
     Not Configured
lnPci (unit number 0):
     Flags: (0x63) UP BROADCAST ARP RUNNING
     Internet address: 172.29.52.88
     Broadcast address: 172.29.255.255
     Netmask 0xffff0000 Subnetmask 0xffff0000
     Ethernet address is 00:00:1a:53:c8:2c
     Metric is 0
     Maximum Transfer Unit size is 1500
     265475 packets received; 18864 packets sent
     0 input errors; 0 output errors
     0 collisions
     Disk IP address: Not Configured Additional Flags: (0x0)
sl (unit number 0):
     Flags: (0x71) UP POINT-TO-POINT ARP RUNNING
     Internet address: 0.0.0.0
     Destination Internet address: 0.0.0.0
     Netmask 0xff000000 Subnetmask 0xff000000
     Metric is 0
     Maximum Transfer Unit size is 576
     0 packets received; 0 packets sent
     0 input errors; 0 output errors
     0 collisions
     Disk IP address: 0.0.0.0
```

dspipifcache

Display IP Interface Cache

The command shows the mapping of SVCs that connect the PXM45s to workstations.

Cards on Which This Command Executes

PXM45

Syntax

```
dspipifcache
[interface]
```

Syntax Description

interface	(Optional) The interface type. If you do not specify an interface type, the display contains cache contents for all interface types. The types are: • **lnPci0** for Ethernet (the default on power-up) • **atm0** for the ATM. • **sl0** for SLIP

Related Commands

dspipif, **ipifconfig**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the contents of the IP interface cache. The display shows that the cache currently is empty.

```
node19.8.PXM.a > dspipifcache
node19-                          System Rev: 02.00   Apr. 07, 2000 16:22:18 PST
MGX8850                                              Shelf Alarm: NONE
IP CONNECTIVITY INTERFACE CACHE
Interface         IpAddress       VcId    Age(Flush@120000)    Flags
--------------------------------------------------------------------------
    No Entries
```

dspndparms

Display Node Parameters—display the current node-level parameters specified by cnfndparms.

The node parameters in this case are a general set of diverse parameters. Refer to **cnfndparms** for a description of the parameters and possible values.

Cards on Which This Command Runs

PXM45

Syntax

dspndparms

Syntax Description

This command takes no parameters.

Related Commands

cnfndparms

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current node parameters.

```
MGX8850.7.PXM.a > dspndparms
MGX8850 System Rev: 02.01   Sep. 04, 2001 12:08:16 PST
MGX8850                                              Node Alarm: CRITICAL
NODE CONFIGURATION OPTIONS
Opt#  Value       Type           Description
----  -----       ----           -----------
1     3600        16bit Decimal  SHM Card Reset Sliding Window (secs)
2     3           8bit Decimal   SHM Max Card Resets Per Window (0 = infinite)
3     Yes         Boolean        Core Redundancy Enabled
4     No          Boolean        Expanded Memory on PXM45B Enabled
5     0x0         8bit Hex       Required Power Supply Module Bitmap
6     0x0         8bit Hex       Required Fan Tray Unit Bitmap
```

dsppvcif

Display PVC Interface

Display details about the PVC interface for IP connectivity. The output shows the:

- Interface type on which the PVC connections exists. Possible types are:
  - ATM (atm0 in the output)
  - Ethernet (lnPcio0 in the output)
  - SLIP interface (sl0 in the output)
- Alarms, if any
- The operational state
- Flags specified for the PVC support (through the **pvcifconfig** command)
- The number of logical connection numbers (LCNs) in the receive and transmit direction)
- Numbers in input and output frames

Cards on Which This Command Executes

PXM45

Syntax

dsppvcif

Syntax Description

This command takes no parameters.

Related Commands

dspipif, pvcifconfig, ipifconfig, dspipifcache

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the current ATM interface state.

```
orioses5.1.PXM.a > dsppvcif
orioses5                         System Rev: 01.00   Aug. 10, 2000 18:36:01 GMT
SES-CNTL                                             Node Alarm: NONE
IP CONNECTIVITY PVC CACHE
--------------------------------------------------------------------
atm (unit number 0):
    Feeder VPI.VCI: 3.8
        Flags:           (0x38) VCMUX,PVC,FEEDER
        State:           (0x1) UP
        RxLCN:           722           TxLCN:            32776
        LCNindex:        0             Feeder Name:      svcbpx16
        Input Frames:    10            Output Frames:    10
        Input Errors:    0             Output Errors:    0
```

dsprevs

Display Revisions—show all firmware versions by physical and logical slot numbers.

The **dsprevs** command shows the current versions of firmware for all slots. The optional parameter lets you see the status of a firmware revision change that is in progress. Slight variations exist in the display with and without the optional status parameter:

- Without a parameter, **dsprevs** shows the versions of current runtime firmware and boot firmware.
- With the optional rev-change status parameter, **dsprevs** shows the current runtime firmware version, the primary and secondary runtime firmware versions, and the status of the revision change. Usually, a revision change is an upgrade rather than a downgrade.

For information on graceful firmware upgrades, see **loadrev**, **runrev**, and so on. Note that the **dspversion** command shows the current and boot firmware versions for only the current slot.

Note The portion of the display that shows slots 17–32 is reserved for future use.

Cards on Which This Command Runs

PXM45

Syntax

```
dsprevs [-status]
```

Syntax Description

-status	Cause the display to show the status of a firmware revision change. If a revision change is in progress in a particular slot, the Rev Change Status column shows the command—**loadrev** for example—for that slot and whether the process is an upgrade ("U") or a downgrade ("D".)

Related Commands

runrev, **loadrev**, **dspversion**, **dspcd**, **dspcds**

Note The **dspcd** and **dspcds** commands show a firmware revision change in progress with a U (for upgrade) or D (for downgrade).

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Display all firmware versions and include status of any firmware upgrades. Note that the display shows the logical slot number 7 for physical slot 8. The display shows that no firmware upgrades are in progress. If an upgrade were in progress on slots 7 and 8 and the present command were loadrev, the Rev Change Status column would show “Loadrev in prog-U.”

```
SanJose.7.PXM.a > dsprevs -s
SanJose                          System Rev: 02.00   Dec. 14, 2000 14:39:05 PST
MGX8850                                              Node Alarm: CRITICAL
Phy. Log. Cur Sw           Prim Sw          Sec Sw           Rev Chg
Slot Slot Revision         Revision         Revision         Status
---- ---- --------         --------         --------         -------

01   01   ---              ---              ---              ---
02   02   ---              ---              ---              ---
03   03   ---              ---              ---              ---
04   04   2.0(123)A1       2.0(123)A1       2.0(123)A1       ---
05   05   ---              ---              ---              ---
06   06   ---              ---              ---              ---
07   07   2.0(123.1)       2.0(123.1)       2.0(123.1)       ---
08   07   ---              2.0(123.1)       2.0(123.1)       ---
09   09   ---              ---              ---              ---
10   10   ---              ---              ---              ---
11   11   2.0(123)A1       2.0(123)A1       2.0(123)A1       ---
12   12   2.0(116.1)       ---              ---              ---
13   13   ---              ---              ---              ---
14   14   ---              ---              ---              ---
15   15   ---              ---              ---              ---
16   16   ---              ---              ---              ---
```

Display revisions.

```
SanJose.7.PXM.a > dsprevs
SanJose                          System Rev: 02.00   Dec. 14, 2000 14:39:59 PST
MGX8850                                              Node Alarm: CRITICAL
Physical  Logical   Inserted       Cur Sw              Boot FW
Slot      Slot      Card           Revision            Revision
--------  -------   --------       --------            --------

01        01        ---            ---                 ---
02        02        ---            ---                 ---
03        03        ---            ---                 ---
04        04        AXSM_16OC3     2.0(123)A1          2.0(106)A1
05        05        ---            ---                 ---
06        06        ---            ---                 ---
07        07        PXM45          2.0(123.1)          2.0(106)A1
08        07        ---            ---                 ---
09        09        ---            ---                 ---
10        10        ---            ---                 ---
11        11        AXSM_16T3E3    2.0(123)A1          2.0(106)A1
12        12        ---            2.0(116.1)          ---
13        13        ---            ---                 ---
14        14        ---            ---                 ---
15        15        ---            ---                 ---
16        16        ---            ---                 ---
17        17        ---            ---                 ---
18        18        ---            ---                 ---
19        19        ---            ---                 ---
20        20        ---            ---                 ---
21        21        ---            ---                 ---
22        22        ---            ---                 ---
23        23        ---            ---                 ---
```

```
24          24          ---               ---                     ---
25          25          ---               ---                     ---
26          26          ---               ---                     ---
27          27          ---               ---                     ---
28          28          ---               ---                     ---
29          29          ---               ---                     ---
30          30          ---               ---                     ---
31          31          ---               ---                     ---
32          32          ---               ---                     ---
```

dspserialif

Display Serial Interface

The **dspserialif** command displays the data rate on one of the serial interfaces on the PXM45-UI-S3 back card. See **cnfserialif** for an explanation. (See the *Cisco MGX 8850 Routing Switch Software Configuration Guide* for an explanation of the application of these physical ports.)

Cards on Which This Command Executes

PXM45

Syntax

```
dspserialif <port#>
```

Syntax Description

port#	Specifies the physical port: • 1=maintenance port. • 2=console port.

Related Commands

addserialif, **delserialif**, **cnfserialif**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display the console port speed.

```
Jupiter_Lower.7.PXM.a > dspserialif 2
  SerialPortNum   : 2
  SerialPortType  : console
  SerialPortSpeed : 19200
```

dspsnmp

Display SNMP Strings

The **dspsnmp** command displays the SNMP strings.

Cards on Which This Card Executes

PXM45

Syntax

`dspsnmp`

Syntax Description

This command takes no parameters.

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

Display the current SNMP strings. This example shows that the only specified string is the community "ro."

```
node19.8.PXM.a > dspsnmp
node19-                      System Rev: 02.00   Apr. 11, 2000 15:04:00 PST
MGX8850                                          Shelf Alarm: NONE

Community:                                           ro
System Location:
System Contact
```

dsptrapmgr

Display Trap Manager

Display details about all existing trap managers. The maximum number of trap managers on a switch is 12. The **dsptrapmgr** output shows:

- IP address of each trap manager
- Port number on the connected work station
- Row status
- Read trap flag stats
- Next trap sequence number

Of these elements, the IP address and port number result from **addtrapmgr**.

Cards on Which This Command Runs

PXM45

Syntax

`dsptrapmgr`

Syntax Description

This command takes no parameters.

Related Commands

addtrapmgr, **deltrapmgr**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display trap managers.

```
node19.8.PXM.a > dsptrapmgr
     ipAddress     PortNum  RowStatus   ReadTrapFlag  NextTrapSeqNum
  ---------------  -------  ----------  ------------  --------------
  171.71.55.21      2500       Add           Off       0
  172.29.65.87      2500       Add           Off       348
  172.71.59.21      2500       Add           Off       0

  LastTrapSeqNum:     385
  NumOfValidEntries:  3
```

dspusers

Display Users

Displays all current users and their access levels if the keyword **-u** is not given. If the key word **-u** is specified, it displays the user ID and access level of that user only.

Cards on Which This Command Runs

PXM45

Syntax

```
dspusers
[-u <userID>]
```

Syntax Description

-u	Keyword that specifies the user (*userId)* to display.

Related Commands

adduser, **deluser**, **cnfuser**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Show all configured users.

```
raviraj.7.PXM.a > dspusers

UserId        AccessLevel
-------------------------
cisco         CISCO_GP
service       SERVICE_GP
superuser     SUPER_GP
```

Show access level for a specified user. The user ID is “raoul.”

```
raviraj.7.PXM.a > dspusers -u raoul

UserId        AccessLevel
-------------------------
raoul          SUPER_GP
```

dspversion

Display Version—display firmware versions on an individual card.

Show details for the versions of boot and runtime firmware residing on a card. Typically, you would use **dspversion** in conjunction with the commands for changing a card's firmware version. (See Related Commands section.) For example, you can use **dspversion** to see if a particular firmware version is currently running.

Version Numbering Conventions

This section describes how to interpret the *version* number of a firmware image. Commands such as **loadrev** and **setrev** require a version number rather than a filename. Similarly, **dspversion** shows the firmware version number rather than the firmware filename. Although the version number derives from the firmware filename, they are distinctly different.

Firmware Filenames

The FW directory on the hard drive contains firmware files of possibly many revisions. (Each firmware file has the *fw* file extension.) The format of a firmware filename is:

cardtype_version-element[_platform].fw

Note that *platform* is an optional field because it applies to only the PXM45. For example, a firmware file may have the name "axsm_002.000.000.001.fw." Within this filename, the version-portion is 002.000.000.001. (Note the absence of "mgx.") This version-portion has the following format:

major-release.minor-release.maintenance.patch

Using the example of axsm_002.000.000.001.fw, the version is 2.0(1.1). Similarly, if no patch were present, the version number would be 2.0(1).

The range for each *release*, *maintenance*, and *patch* is 0–255. Note that, as you read left-to-right, each element is a superset of the element on the right, and the number on the right resets to 0 or 1 when the element on its *left* is incremented. For example, if the *minor-release* number 010 rolls to 011, the *maintenance* on its right is reset to 1, so the new version in the example is 002.011.000.000. (Note the anomaly here is that the *maintenance* number resets to 1 rather than 0 because of the IOS convention of starting maintenance numbers at 1.)

Version Numbers

To derive the firmware version number, the firmware filename is altered by removing insignificant zeroes and being reformatted to include parentheses. The format of a *version* number is:

major-release.minor-release(maintenance.patch)phase

For example, the significance of 2.0(14.8)P1 is shown below:

major-release	minor-release	(maintenance.patch)	phase
2.	1.	(6.0)	P1

Pre-release, developmental versions have one or two alphanumeric characters at the end of the version number, and these versions may appear in various contexts. For example, the help display for **setrev** gives examples of *revision*, but only the first two in the following list could be in *released* product. These two bullets show major release 2, minor release 1, and the minimal maintenance number of 1. The last three bullets show developmental revision numbers:

- 2.0(1) Note the absence of a patch number.
- 2.0(1.248) Note the patch number is 248.
- 2.0(0.1)A1 Note the phase number is A1.
- 2.0(0.10)D2 Note the phase number is D2.
- 2.0(0.248)P1, 2.0(0.1)P2, 2.0(0.113)P3, 2.0(0.10)P4

Cards on Which This Command Executes

PXM45, AXSM

Syntax

dspversion

Syntax Description

This command takes no parameters.

Related Commands

abortrev, **commitrev**, **loadrev**, **runrev**, **setrev**, **dspcd**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the firmware version for the current PXM45.

```
pop20two.7.PXM.a > dspversion

Image Type   Shelf Type    Card Type         Version   Built On
  ----------   ----------   ----------    ------------   ------------
     Runtime          MGX        PXM45     2.0(0.30)P2   Jan 17 2001, 16:43:13
        Boot          MGX        PXM45     2.0(0.30)P1   -
```

Display firmware image on the AXSM in slot 1. As the example shows, the command executes on the CLI of the AXSM after you have switched (**cc**) to that CLI.

```
Unknown.1.AXSM.a > dspversion

Image Type   Shelf Type    Card Type         Version   Built On
  ----------   ----------   ----------    ------------   ------------
     Runtime          MGX         AXSM     2.0(0.30)P2   Jan 17 2001, 17:06:31
        Boot          MGX         AXSM      2.0(0.15)A   -
```

ipifconfig

IP Interface Configuration

Configure an interface to provide IP connectivity for user-control of the switch. Typically, the Cisco WAN Manager application running on a local or remote work station uses this connection to control the switch.

(Note that **ipifconfig** and related commands have no bearing on the Console Port for an ASCII terminal that is co-located with the node. For details on the hardware connections and initial start-up through the console port, see the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2 and *Cisco MGX 8850 Hardware Installation*, Release 2.)

The **ipifconfig** command lets you specify:

- A type of logical interface—ATM, Ethernet, or SLIP
- An IP address for the interface
- A broadcast address for Ethernet
- Restriction of connections to be either SVCs or PVCs
- Support at the interface level for ARP
- An interface type that serves as the default when the switch powers up
- The operational state of the interface—up or down

Usage Guidelines

Except for the first-time, mandatory configuration of an IP address for the interface, the **ipifconfig** parameters are optional details that you can use to modify the interface. The design of the parameters includes default states that apply to a broad range of network designs. The purpose of this default design is to minimize the need to change the optional parameters.

The IP interface configuration requires knowledge of the capabilities of the devices or interfaces that exist between the PXM45 and workstation. Particularly, any attached routers should be feature-rich. For example, the most likely configuration consists of:

- Cisco 7000-family or Cisco 4500-series routers connected to an AXSM port. An MGX 8850 switch can communicate with a maximum of 42 routers. If one router does not, for example, support ATMARP, you must configure the interface not to use ATMARP when communicating with any router in the IP connectivity path.
- An ATM router interface with LLC encapsulation and ATMARP protocol service, RIP protocol, and ILMI protocol.
- A Sun workstation running Cisco WAN Manager with support for RIP and SNMP protocols.

Note The **ipifconfig** command on the PXM45 corresponds to **cnfifip** on the PXM1.

Cards on Which This Command Runs

PXM45

Syntax

```
ipifconfig <interface>
[ip_address]
[netmask <mask>]
[broadcast <broad_addr>]
[up | down ]
[arp | noarp]
[svc | nosvc]
[pvc | nopvc]
[default | nodefault]
[clrstats]
```

Syntax Description

interface	A name that identifies the type of interface. The type of interface affects the applicability of other **ipifconfig** parameters. The choices for *interface* are: • **lnPci0** for Ethernet (the default on power-up) • **atm0** for the ATM • **sl0** for SLIP
ip_address	(Optional if already configured, mandatory if not) *ip_address* is a 32-bit IP address in dotted decimal format. This parameter is mandatory when you first configure a particular interface type (**lnPci0**, and so on). If you subsequently modify one or more optional interface parameters, you can omit this IP address because the interface name (*interface,* above) is sufficient to get the address.
netmask	(Optional) 32-bit net mask in dotted decimal format. Ideally, the PXM45 and any routers associated with connected workstations exist in the same subnet. Specifically, having the same subnet mask simplifies router configuration.
broadcast	(Optional) Broadcast address—applies to only Ethernet.
up | down	(Optional) Set the interface to be either up or down. Default is **up**. Setting it to **down** turns off all IP packet communication. You should have a specific purpose for downing the interface.
arp | noarp	(Optional) Enables or disables ARP for all connections on the interface. Enter the keyword **arp** or **noarp** in its entirety. The default is enabled (**arp**). Note that *disabling* ARP for Ethernet is a very unlikely choice. If you disable ARP, the system subsequently prevents you from specifying ARP for an individual SVC or PVC. If you need to disable ARP for a connection because a particular interface or device does not support ARP, disable it though **svcifconfig** or **pvcifconfig**.
svc | nosvc	(Optional) Specify whether SVC support is enabled on the interface. The choice applies to all connections on the interface. The default is enabled (**svc**) and is the most common application. Specific contexts may provide a reason to disable SVCs on the interface.

pvc | nopvc	(Optional) Specify whether PVC support is enabled. The default for this parameter is enabled (**pvc**). The application of PVC support is for a device in the network management path that provides IP connectivity but does not support SVCs. With PVC support enabled, you subsequently set up a PVC to that device by executing **pvcifconfig**. If PVC support is not enabled, **pvcifconfig** fails. If you change this value, type the word **pvc** or **nopvc** in its entirety.
default | nodefault	(Optional) Specifies whether to use this interface as the default interface. As stated in the description of the *interface* parameter, the default interface is Ethernet the first time the switch powers up. You can change the default by entering the **default** or **nodefault** keyword. For example, if you currently are specifying an ATM interface (**atm0**) on the control port, you can make it the default (upon subsequent node reset) by typing the keyword **default**.
clrstats	Clear all interface and connection statistics for the specified *interface* type. The statistics pertain to incoming and outgoing packets, errored packets, and so on.

Related Commands

dspipif, pvcifconfig, dsppvcif, svcifconfig, **dspsvcif**, **dspipifcache**

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

Specify a IP interface with an ATM interface type, address of 163.72.29.177, and a net mask of 255.255.255.000, and use the defaults for all other parameters.

```
wilco.7.PXM.a > ipifconfig atm0 163.72.29.177 mask 255.255.255.000
```

loadrev

Load Revision

Downloads a firmware image from the FW directory on the disk to flash memory on the targeted card. Executing is the first step in performing a graceful firmware upgrade. A graceful revision change preserves the configuration of the card and minimizes any data loss that could result from the brief disruption in service.

Although **loadrev** runs on a PXM45, the target can be either a service or the PXM45 itself. The system automatically determines which card in a redundant setup is active and which is standby. Specifying the active card slot is sufficient. For example, if a PXM45 is the target, you can specify either slot 7 or slot 8 regardless of the active slot number.

The sequence of commands for a graceful revision change appear in the following list. See Table 2-10 and Table 2-11 for a clarification of the various states within this sequence.

1. **loadrev** loads a firmware version from the hard disk to a card's memory. In a non-redundant card setup, **loadrev** does not cause the system to reset the card.
2. **runrev** causes the primary card to start running the new version. For a redundant pair of cards, the standby becomes the active card then starts running the new version.
3. If an unacceptable problem occurs, the optional **abortrev** command restores the previous version of firmware as well as the previous database contents.
4. **commitrev** declares the new primary version to be acceptable and removes the old primary from main memory (but not the hard disk).

A graceful upgrade takes a single card or a redundant card pair through different stages. Also, if you must execute **abortrev** on a redundant pair, the card (or possibly both cards in a redundant pair) are reset. The stages of a graceful upgrade and the reset actions appear in Table 2-10 for a single-card upgrade and Table 2-11 for a redundant-pair upgrade.

The tables start by showing that, initially, the primary and secondary versions of firmware are 2.x, so the only possible operational version is 2.x. The **loadrev** command loads a generic version called 2.y, and the upgrade sequence progressively changes the primary and secondary firmware versions.

Table 2-10 Single-Card Upgrade From 2.x to 2.y

Firmware Status	Initial Version	After loadrev	After runrev	After commitrev
Primary	2.x	2.x	2.y	2.y
Secondary	2.x	2.y	2.x	2.y
Operational	2.x	2.x	2.y	2.y
			After **abortrev**, the card is reset.	

Note Of special note in Table 2-11, **runrev** causes the standby card to become the active card. The reversed location of the "Active" and "Standby" columns shows the changed states.

Table 2-11 Redundant Pair Upgrade From 2.x to 2.y

Firmware status	Before upgrade		After loadrev		After runrev		After commitrev	
	Active	Standby	Active	Standby	Standby	Active	Standby	Active
Primary	2.x	2.x	2.x	2.x	2.y	2.y	2.y	2.y
Secondary	2.x	2.x	2.y	2.y	2.x	2.x	2.y	2.y
Current	2.x	2.x	2.x	2.y	2.y	2.y	2.y	2.y
			abortrev resets only standby card.		**abortrev** resets both cards.			

After you execute **runrev**, the PXM45 updates the database records on disk if changes occur (such as changes to the configuration or network topology). If you revert to the previous version by executing **abortrev**, the post-**runrev** changes are lost. For example, if a switch was added to the network between **runrev** and **abortrev**, the restored database has no record of the topology change.

Version Numbering Conventions

This section describes how to interpret the *version* number of a firmware image. Commands such as **loadrev** and **setrev** require a version number rather than a filename. Similarly, **dspversion** shows the firmware version number rather than the firmware filename. Although the version number derives from the firmware filename, they are distinctly different.

Firmware Filenames

The FW directory on the hard drive contains firmware files of possibly many revisions. (Each firmware file has the *fw* file extension.) The format of a firmware filename is:

cardtype_version-element[_platform].fw

Note that *platform* is an optional field because it applies to only the PXM45. For example, a firmware file may have the name "axsm_002.000.000.001.fw." Within this filename, the version-portion is 002.000.000.001. (Note the absence of "mgx.") This version-portion has the following format:

major-release.minor-release.maintenance.patch

The range for each *release*, *maintenance*, and *patch* is 0–255. Note that, as you read left-to-right, each element is a superset of the element on the right, and the number on the right resets to 0 or 1 when the element on its *left* is incremented. For example, if the *minor-release* number 010 rolls to 011, the *maintenance* on its right is reset to 1, so the new version in the example is 002.011.000.000. (Note the anomaly here is that the *maintenance* number resets to 1 rather than 0 because of the IOS convention of starting maintenance numbers at 1.)

Version Numbers

To derive the firmware version number, the firmware filename is altered by removing insignificant zeroes and being reformatted to include parentheses. The format of a *version* number is:

major-release.minor-release(maintenance.patch)phase

For example, the significance of 2.0(14.8)P1 is shown below:

major-release	minor-release	(maintenance.patch)	phase
2.	1.	(6.0)	P1

Pre-release, developmental versions have one or two alphanumeric characters at the end of the version number, and these versions may appear in various contexts. For example, the help display for **setrev** gives examples of *revision*, but only the first two in the following list could be in *released* product. These two bullets show major release 2, minor release 1, and the minimal maintenance number of 1. The last three bullets show developmental revision numbers:

- 2.0(1) Note the absence of a patch number.
- 2.0(1.248) Note the patch number is 248.
- 2.0(0.1)A1 Note the phase number is A1.
- 2.0(0.10)D2 Note the phase number is D2.
- 2.0(0.248)P1, 2.0(0.1)P2, 2.0(0.113)P3, 2.0(0.10)P4

Cards on Which This Command Runs

PXM45

Syntax

```
loadrev <slot> <revision>
```

Syntax Description

slot	The number of the targeted card slot.
revision	Revision number derived from the firmware file. See “Version Numbering Conventions.”

Related Commands

abortrev, **commitrev**, **runrev**, **setrev**, **dspversion**, **dspcd**

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

Load version 2.0(0.4) to the AXSM in slot 5.

```
pinnacle.7.PXM.a > loadrev 5 2.0(0.4)
```

pvcifconfig

PVC Interface Configuration

Modifies an existing PVC to support IP connectivity to a feeder such as a BPX switch or an MGX Release 1 switch.

Cards on Which This Command Runs

PXM45

Syntax

```
pvcifconfig <interface> <router | local> <pvc_address>
[ atmarp | noatmarp ]
[ llcencap | vcmux ]
[ default | nodefault]
[ reset ]
[ delete ]
[ up ]
[ clrstats ]
```

Syntax Description

interface	An alphanumeric string that identifies the interface type. The choices are: **lnPci0** for Ethernet (the default on power-up) **atm0** for the ATM. **sl0** for SLIP Enter the entire keyword.Where appropriate, each subsequent parameter description identifies characteristics that depend on the type of interface.
router | local	Specifies whether the AESA corresponds to a router or the local PXM45. Both router and local ends should be configured. Configure the local end first, then execute **pvcifconfig** to specify the router end. You must enter the entirety of one of these keywords. The AESA is an NSAP address used by the router or the local PXM45.
pvc_address	The VPI and VCI of the PVC. The format is *vpi.vci*.
[atmarp | noatmarp]	(Optional) Enables or disables ATMARP on a PVC—if the connected router supports ATMARP. Furthermore, it applies to only the ATM End Station Address (AESA) configuration at the router's interface. (See **ipifconfig** description.)
llcencap | vcmux	Applies to the router link only. This parameter specifies encapsulation. The choice primarily depends on whether the router supports LLC Snap encapsulation (**llcsnap**). The alternative is VC-based multiplexing (**vcmux**).
default | nodefault	(Optional) Specifies whether this PVC is the default route on the interface.

reset	(Optional) Resets the PVC.
delete	(Optional) Delete the specified AESA configuration.
clrstats	(Optional) Clear all SVC statistics on this interface.
up	(Optional) Put the PVC in the UP state and try to bind the associated lcns.
default | nodefault	(Optional) Specifies whether this PVC is the default route on the interface.
clrstats	(Optional) Clear any statistics for this PVC (dropped packets, for example).

Related Commands

dsppvcif, **ipifconfig**, **setipconndebug**

Attributes

Log: no log	State: active	Privilege: SUPER_GP

resetsys

Reset System

Reset the entire switch.

Cards on Which This Command Runs

PXM45

Syntax

`resetsys`

Syntax Description

This command takes no parameters but displays a warning and prompts you to continue the execution.

Related Commands

resetcd

Attributes

Log: no log	State: active, init	Privilege: SUPER_GP

Example

Reset the system. When prompted to confirm this action, say no.

```
pinnacle.7.PXM.a > resetsys
This command resets the entire shelf, a destructive command.
Please confirm now!
Do you want to proceed (Yes/No)? no
(command not executed)
```

restoreallcnf

Restore All Configurations

Restores all configuration files saved to the CNF directory on the hard drive. The saved configuration is the result of a prior execution of the **saveallcnf** command. To see a list of existing configurations that have been zipped by **saveallcnf**, **cd** to the C drive and list the contents of the CNF directory.

Cards on Which This Command Runs

PXM45

Syntax

restoreallcnf -f *filename* [-v]

Syntax Description

-f	Specifies the filename of the zipped version of the configuration.
-v	(Optional) Specifies that a list of the restored configuration files goes to the default printer. The default is no printout.

Related Commands

saveallcnf

Attributes

Log: no log	State: active, init	Privilege: SERVICE_GP

Example

Restore the system configuration created on February 5, 2001. The system responds with a statement of what the command would do and prompts you to accept the action before it proceeds with command execution.

```
pinnacle.7.PXM.a > restoreallcnf -f pop20two_01_200102051156.zip
The current configuration will be replaced with the contents
of the specified file and the shelf will be rebooted.

restoreallcnf: Do you want to proceed (Yes/No)? n
(command not executed
```

runrev

Run Revision

Causes a new firmware version to start running. In a redundant card pair, **runrev** first causes the standby card to become the active card. The **runrev** command is the second of the required commands in a graceful upgrade. It runs on the PXXM45 but can target either a service module or the PXM45.

The order of commands in a graceful upgrade, including the option of aborting the upgrade, appears in the following list. For clarification of the states in a graceful upgrade, see Table 2-12 and Table 2-13.

1. **loadrev** loads a firmware version from the hard disk to a card's memory. In a non-redundant card setup, **loadrev** does not cause the system to reset the CARD.
2. **runrev** causes the primary card to start running the new version. For a redundant pair of cards, the standby becomes the active card then starts running the new version.
3. If an unacceptable problem occurs, the optional **abortrev** command restores the previous version of firmware as well as the previous database contents.
4. **commitrev** declares the new primary version to be acceptable and removes the old primary from main memory (but not the hard disk).

The sequence of commands for a graceful revision change appear in the following list. See Table 2-10 and Table 2-11 for a clarification of the various states within this sequence.

1. **loadrev** loads a firmware version from the hard disk to a card's memory. In a non-redundant card setup, **loadrev** does not cause the system to reset the card.
2. **runrev** causes the primary card to start running the new version. For a redundant pair of cards, the standby becomes the active card then starts running the new version.
3. If an unacceptable problem occurs, the optional **abortrev** command restores the previous version of firmware as well as the previous database contents.
4. **commitrev** declares the new primary version to be acceptable and removes the old primary from main memory (but not the hard disk).

A graceful upgrade takes a single card or a redundant card pair through different stages. Also, if you must execute **abortrev** on a redundant pair, the card (or possibly both cards in a redundant pair) are reset. The stages of a graceful upgrade and the reset actions appear in Table 2-10 for a single-card upgrade and Table 2-11 for a redundant-pair upgrade.

The tables start by showing that, initially, the primary and secondary versions of firmware are 2.x, so the only possible operational version is 2.x. The **loadrev** command loads a generic version called 2.y, and the upgrade sequence progressively changes the primary and secondary firmware versions.

Table 2-12 Single-Card Upgrade From 2.x to 2.y

Firmware Status	Initial Version	After loadrev	After runrev	After commitrev
Primary	2.x	2.x	2.y	2.y
Secondary	2.x	2.y	2.x	2.y
Operational	2.x	2.x	2.y	2.y
			After **abortrev**, the card is reset.	

> **Note** Of special note in Table 2-13, **runrev** causes the standby card to become the active card. The reversed location of the "Active" and "Standby" columns shows the changed states.

Table 2-13 Redundant Pair Upgrade From 2.x to 2.y

Firmware status	Before upgrade		After loadrev		After runrev		After commitrev	
	Active	**Standby**	**Active**	**Standby**	**Standby**	**Active**	**Standby**	**Active**
Primary	2.x	2.x	2.x	2.x	2.y	2.y	2.y	2.y
Secondary	2.x	2.x	2.y	2.y	2.x	2.x	2.y	2.y
Current	2.x	2.x	2.x	2.y	2.y	2.y	2.y	2.y
			abortrev resets only standby card.		**abortrev** resets both cards.			

After you execute **runrev**, the PXM45 updates the database records on disk if changes occur (such as changes to the configuration or network topology). If you revert to the previous version by executing **abortrev**, the post-**runrev** changes are lost. For example, if a switch was added to the network between **runrev** and **abortrev**, the restored database has no record of the topology change.

Cards on Which This Command Runs

PXM45

Syntax

runrev
<*slot*>
<*revision*>

Syntax Description

slot	Number of the targeted card slot.
revision	Revision number derived from the name of the firmware file. If the standby card does not have the specified image, **runrev** has no effect, and the system displays an error message. For an explanation, see the section, "Version Numbering Conventions," in the **loadrev** description.

Related Commands

abortrev, **commitrev**, **loadrev**, **setrev**, **dspcd**, **dspversion**

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

Run version 2.0(0.4) in logical slot 7. A previous check of the cards (by using **dspcds**) and firmware images (by using **dspcd**) would show whether a redundant card and version 2.0(0.4) are present.

```
excel.8.PXM.a > runrev 7 2.0(0.4)
```

routeShow

Route Show

Show the current IP routing of the network layer of the operating system.

Cards on Which This Command Runs

PXM45

Syntax

routeShow

Related Commands

routestatShow

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the current IP routing of the network layer of the operating system.

```
pinnacle.8.PXM.a > routeShow

ROUTE NET TABLE
destination      gateway          flags  Refcnt  Use           Interface
-----------------------------------------------------------------------
0.0.0.0          172.29.23.149    1      1       21778         lnPci0
0.0.0.0          172.29.23.1      3      0       2755          lnPci0
172.1.1.0        172.1.1.149      1      0       0             atm0
172.29.23.0      172.29.23.149    1      2       5275          lnPci0
-----------------------------------------------------------------------

ROUTE HOST TABLE
destination      gateway          flags  Refcnt  Use           Interface
-----------------------------------------------------------------------
0.0.0.0          0.0.0.0          5      0       0             sl0
127.0.0.1        127.0.0.1        5      1       0             lo0
172.29.23.3      172.1.1.149      5      0       3555          atm0
172.29.23.5      172.1.1.149      5      0       3304          atm0
172.29.23.7      172.1.1.149      5      0       3335          atm0
171.71.29.18     172.1.1.149      5      0       3304          atm0
172.29.23.18     172.1.1.149      5      0       3304          atm0
172.29.23.28     172.1.1.149      5      0       6127          atm0
172.29.23.29     172.1.1.149      5      1       6065          atm0
171.71.29.32     172.1.1.149      5      0       5842          atm0
171.71.29.44     172.1.1.149      5      0       3304          atm0
172.29.23.53     172.1.1.149      5      0       3304          atm0
171.71.29.59     172.1.1.149      5      0       3304          atm0
171.71.28.126    172.1.1.149      5      0       3309          atm0
-----------------------------------------------------------------------

pinnacle.8.PXM.a >
```

routestatShow

Show Routing Statistics

Use the **routestatShow** command to view the current IP routing statistics for the network layer of the operating system.

Cards on Which This Command Runs

PXM45

Syntax

routestatShow

Related Commands

routeShow

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the current IP routing statistics for the network layer of the operating system

```
pinnacle.8.PXM.a > routestatShow

routing:
        0 bad routing redirect
        0 dynamically created route
        0 new gateway due to redirects
        0 destination found unreachable
        11095 uses of a wildcard route

pinnacle.8.PXM.a >
```

saveallcnf

Save All Configurations—save all configuration files to the hard drive.

The **saveallcnf** command saves all configurations to a zipped file in the CNF directory on the hard drive. This command takes significant time, so a warning message prompts you for confirmation before the system performs the task. Upon completion, the system displays the name of the saved configuration file. The system stores up to two zipped configuration files. If you want to save more than two configurations, use FTP to transfer the files to another device.

To restore the system configuration, use **restoreallcnf**.

Note You should execute **saveallcnf** only if no connection provisioning is occurring.

Cards on Which This Command Runs

PXM45

Syntax

```
saveallcnf
```

Related Commands

restoreallcnf

Attributes

Log: log	State: active, init	Privilege: SERVICE_GP

Example

Save the system configuration. Note that the system displays the name of the saved configuration file.

```
MGX8850.7.PXM.a > saveallcnf
The 'saveallcnf' command can be time-consuming. The shelf
must not provision new circuits while this command is running.

Do not run this command unless the shelf configuration is stable
or you risk corrupting the saved configuration file.


saveallcnf: Do you want to proceed (Yes/No)? n
(command not executed)
```

setipconndebug

Set IP Connection Debug

Specify a debug mode and whether to use a console for debugging IP connectivity. This command requires SUPER_GP privilege. After you set the debug level, a status message states the current level.

Syntax

setipconndebug [**-console** | **-noconsole**] [*debuglevel*]

Syntax Description

-console | -no console	Configure where to run **setipconndebig:** console (ASCII) terminal or elsewhere.
debuglevel	Specifies a debug level. To select one or all of the following levels, enter the associated hexadecimal number and include the leading “0x” (see example): • No Logging (0x0) • Task Errors (0x1) • Task Debug (0x2) • ATM Protocol (0x4) • Task Startup (0x8) • Task Events (0x10) • SVC Call Events (0x20) • ATMARP Protocol (0x40) • Task Timers (0x80) • Interface Cache (0x100) • Subtask Events (0x200) • DISKDB Events (0x400) • RAMDB Events (0x800) • TRAP Events (0x1000) • All Logging (0xffffffff)

Attributes

Log: no log	State: active, standby, init	Privilege: SERVICE_GP

Example

Set IP connection debug to console and specify a debug level of 20 for SVC call events.

```
node19.8.PXM.a > setipconndebug -console 20
```

setrev

Set Revision

Force-load and run a firmware version for a card. You must execute **setrev** from the CLI of the active PXM45 whether the target is a service module or the PXM45.

Note For the first-time power-up of the node, you should execute the **burnboot** command to burn in the bootcode. For details, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.

From a high-level perspective, the **setrev** command has two effects. It causes the PXM45 to load a firmware image from the hard drive to a card, then it causes the receiving card to run that image. The impact is a non-graceful revision change. (A graceful revision path is available through the sequence of **loadrev**, **runrev**, and **commitrev**. A revision is an *upgrade* if the new firmware version has a higher numerical value or a *downgrade* if the new version has a lower value.)

At the time you initially bring up a node or after executing **clrallcnf**, the service modules have no runtime firmware image, so you must execute **setrev** for each service module in the switch. For the PXM45, Cisco ships the product with firmware installed, so executing **setrev** is not necessary until you need to change firmware version or after you execute **clrallcnf**.

Version Numbering Conventions

This section describes how to interpret the *version* number of a firmware image. Commands such as **loadrev** and **setrev** require a version number rather than a filename. Similarly, **dspversion** shows the firmware version number rather than the firmware filename. Although the version number derives from the firmware filename, they are distinctly different.

Firmware Filenames

The FW directory on the hard drive contains firmware files of possibly many revisions. (Each firmware file has the *fw* file extension.) The format of a firmware filename is:

cardtype_version-element[_platform].fw

Note that *platform* is an optional field because it applies to only the PXM45. For example, a firmware file may have the name “axsm_002.000.000.001.fw.” Within this filename, the version-portion is 002.000.000.001. (Note the absence of “mgx.”) This version-portion has the following format:

major-release.minor-release.maintenance.patch

The range for each *release*, *maintenance*, and *patch* is 0–255. Note that, as you read left-to-right, each element is a superset of the element on the right, and the number on the right resets to 0 or 1 when the element on its *left* is incremented. For example, if the *minor-release* number 010 rolls to 011, the *maintenance* on its right is reset to 1, so the new version in the example is 002.011.000.000. (Note the anomaly here is that the *maintenance* number resets to 1 rather than 0 because of the IOS convention of starting maintenance numbers at 1.)

Version Numbers

To derive the firmware version number, the firmware filename is altered by removing insignificant zeroes and being reformatted to include parentheses. The format of a *version* number is:

major-release.minor-release(maintenance.patch)

Using the example of axsm_002.000.000.001.fw, the version is 2.0(1.1). Similarly, if no patch were present, the version number would be 2.0(1).

Pre-release, developmental versions have one or two alphanumeric characters at the end of the version number, and these versions may appear in various contexts. For example, the help display for **setrev** gives examples of *revision*, but only the first two in the following list could be in *released* product. These two bullets show major release 2, minor release 1, and the minimal maintenance number of 1. The last three bullets show developmental revision numbers:

- 2.0(1) Note the absence of a patch number.
- 2.0(1.248) Note the patch number is 248.
- 2.0(0.1)A1 Note the phase number is A1.
- 2.0(0.10)D2 Note the phase number is D2.
- 2.0(0.248)P1, 2.0(0.1)P2, 2.0(0.113)P3, 2.0(0.10)P4

Note The **setrev** command resets the active PXM45 only if the revision changes on the active card are a result of the **setrev** command.

Cards on Which This Command Runs

PXM45

Syntax

setrev *<slot> <version>*

Note With the current release, the primary and secondary images are the same. For this reason, you do not specify the secondary revision, so the syntax line indicates only “version.”

Syntax Description

slot	Slot number of the card targeted for firmware specification.
version	An alphanumeric string derived from the name of the firmware file. For an explanation of the numbering scheme, see the section, “Version Numbering Conventions,” earlier in the **setrev** description.

Related Commands

loadrev, **runrev**, **commitrev**, **abortrev**, **dspversion**, **dspcd**

Attributes

Log: log	State: active	Privilege: SERICE_GP

Example

Specify version 2.0(2) for the card in slot 9. In addition to **setrev**, this example shows other commands you could use before and after **setrev**. The sequence begins with a display of all the cards. While the firmware is going into the RAM on the card, periodically execute **dspcds** on the PXM45 to see the changing status of the target card. After **setrev** finishes, execute **dspcd** on the targeted service module to see the version and other details of the card or dspversion to see just the version.

Step 1 On the PXM45, use **dspcds**. The display shows slot 9 has a card with no firmware.

```
pinnacle.7.PXM.a > dspcds
pxm45tl                          System Rev: 00.00   Jan. 05, 2000 15:18:40 GMT
Boot F/W Rev: 0.0(0)             H/W Rev:    00.00   GMT Offset 0
Backplane Serial No: _UNKNOWN___ Backplane HW Rev: 00.00
Statistics Master IP Address: 0.0.0.0                  Shelf Alarm: NONE
Card  Front/Back       Card           Alarm      Redundant  Redundancy
Slot  Card State       Type           Status     Slot       Type
---   ----------       --------       --------   -------    -----

01    Empty            ---            ---        ---         ---
02    Empty            ---            ---        ---         ---
03    Empty            ---            ---        ---         ---
04    Empty            ---            ---        ---         ---
05    Empty            ---            ---        ---         ---
06    Empty            ---            ---        ---         ---
07    Active/Empty    UNKNOWN_FC      NONE       08         PRIMARY SLOT
08    Empty Resvd/Emp UNKNOWN_FC      MAJOR      07         SECONDARY SLOT
09    Failed/Empty    UNKNOWN_FC      NONE       NA         NO REDUNDANCY
10    Empty            ---            ---        ---         ---
11    Empty            ---            ---        ---         ---
12    Empty            ---            ---        ---         ---
13    Empty            ---            ---        ---         ---
14    Empty            ---            ---        ---         ---
```

Step 2 Change directories to the “FW” (firmware) directory.

```
pinnacle.7.PXM.a > cd /FW
```

Step 3 List the contents of the directory:

```
pinnacle.7.PXM.a > ls
```

The display shows the names of the firmware files. Extract the AXSM version number—2.0(2):

```
pxm45_002.000.001-D.fw
pxm45_002.000.014-A1_bt.fw
axsm_002.000.002.fw
```

Step 4 Type **setrev** and specify version 2.0(2) as the primary firmware version for slot 9.

```
pinnacle.7.PXM.a > setrev 9 2.0(2)
```

Step 5 Check the progress by executing **dspcds**. The following display shows that the PXM45 has detected the card type in slot 9. The status is “init”—initialization in progress:

```
pxm45tl                          System Rev: 00.00   Jan. 05, 2000 15:21:01 GMT
Boot F/W Rev: 0.0(0)             H/W Rev:    00.00   GMT Offset  0
Backplane Serial No: _UNKNOWN___ Backplane HW Rev: 00.00
Statistics Master IP Address: 0.0.0.0                  Shelf Alarm: NONE
Card  Front/Back       Card           Alarm      Redundant  Redundancy
Slot  Card State       Type           Status     Slot       Type
---   ----------       --------       --------   -------    -----

01    Empty            ---            ---        ---         ---
02    Empty            ---            ---        ---         ---
03    Empty            ---            ---        ---         ---
04    Empty            ---            ---        ---         ---
05    Empty            ---            ---        ---         ---
06    Empty            ---            ---        ---         ---
07    Active/Empty    UNKNOWN_FC      NONE       08         PRIMARY SLOT
08    Empty Resvd/Emp UNKNOWN_FC      MAJOR      07         SECONDARY SLOT
09    Init/Empty      AXSM_16OC3      NONE       NA         NO REDUNDANCY
10    Empty            ---            ---        ---         ---
11    Empty            ---            ---        ---         ---
12    Empty            ---            ---        ---         ---
13    Empty            ---            ---        ---         ---
14    Empty            ---            ---        ---         ---
```

Step 6 The next execution of **dspcds** indicates the card is active. Therefore, the firmware is running.

```
pxm45tl                          System Rev: 00.00   Jan. 05, 2000 15:21:11 GMT
Boot F/W Rev: 0.0(0)             H/W Rev:    00.00   GMT Offset  0
Backplane Serial No: _UNKNOWN___ Backplane HW Rev: 00.00
Statistics Master IP Address: 0.0.0.0                  Shelf Alarm: NONE
Card  Front/Back       Card           Alarm      Redundant  Redundancy
Slot  Card State       Type           Status     Slot       Type
---   ----------       --------       --------   -------    -----

01    Empty            ---            ---        ---         ---
02    Empty            ---            ---        ---         ---
03    Empty            ---            ---        ---         ---
04    Empty            ---            ---        ---         ---
05    Empty            ---            ---        ---         ---
06    Empty            ---            ---        ---         ---
07    Active/Empty    UNKNOWN_FC      NONE       08         PRIMARY SLOT
08    Empty Resvd/Emp UNKNOWN_FC      MAJOR      07         SECONDARY SLOT
09    Active/Active   AXSM_16OC3      NONE       NA         NO REDUNDANCY
10    Empty            ---            ---        ---         ---
11    Empty            ---            ---        ---         ---
12    Empty            ---            ---        ---         ---
```

Step 7 Execute **dspversion** to see the version of the runtime image.

```
pinnacle.9.AXSM.a > dspversion

  Image Type   Shelf Type    Card Type         Version   Built On
  ----------   ----------   ----------    ------------   ------------
     Runtime          MGX         AXSM         2.0(2)   Jan 03 2000, 16:36:39
        Boot          MGX         AXSM      2.0(128)A1   -
```

svcifconfig

SVC Interface Configure

Configure IP-related parameters for the SVCs that support network control at a workstation.The configuration applies to all the SVCs on one of the three physical port types. Note that a complete configuration requires you to execute **svcifconfig** twice. The first execution identifies the ATM end-station address (AESA) and encapsulation type at the router end. The second execution identifies the AESA—but no encapsulation type—for the switch.

Cards on Which This Command Runs

PXM45

Syntax

```
svcifconfig <interface> <router | local> <svc_address>
[atmarp | noatmarp]
[llcencap | vcmux]
[default | nodefault]
[reset]
[delete]
[force]
[clrstats]
```

Syntax Description

Enter all keywords in their entirety.

interface	Alphanumeric string identify the interface type for the current SVC configuration. The choices are: **lnPci0** for Ethernet (the default on power-up) **atm0** for the ATM. **sl0** for SLIP Enter the entire keyword.Where appropriate, each subsequent parameter description identifies characteristics that depend on the type of interface.
router | local	Specifies whether the AESA corresponds to a router or the local PXM45. Both router and local ends should be configured. Configure the local end first, then execute **svcifconfig** to specify the router end. You must enter the entirety of one of these keywords. The AESA is an NSAP address used by the router or the local PXM45.
svc_address	The NSAP portion for the SVCs that the switch sets up on the specified interface type.
atmarp | noatmarp	(Optional) This parameter is valid for router AESA configuration only. Enables or disables ATMARP. For ATMARP to be available, the interface must support ARP (see **ipifconfig** description).

llcencap | vcmux	Applies to the router link only. This parameter specifies encapsulation. The choice primarily depends on whether the router supports LLC Snap encapsulation (**llcsnap**). The alternative is VC-based multiplexing (**vcmux**).
default | nodefault	(Optional) Specifies whether this SVC is the default route on the interface.
reset	(Optional) Reset of the SVC. The SVC is freed, then the call is attempted again.
delete	(Optional) Delete the specified AESA configuration.
force	(Optional) Force the SVC to be reset or deleted. Use the force option with reset or delete if reset or delete appears to be hung.
clrstats	(Optional) Clear all SVC statistics on this interface.

Related Commands

ipifconfig, **dspipif**, **dspsvcif**, **dspipifcache**

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

First configure the AESA for the local (PXM45) side, then configure the AESA for the router. This case uses the defaults for encapsulation (llcencap) and ARP (enabled).

```
sfo.7.PXM.a > svcifconfig arm0 local 47.0091.8100.0000.1010.1010.1010.1010.1010.1010.10

sfo.7.PXM.a > svcifconfig arm0 router 47.0091.8100.0000.0101.0101.0101.0101.0101.0101.01
```

switchcc

Switch Core Cards

Switch control of the MGX 8850 node from the present slot to the other PXM45 slot. If a standby PXM45 is not available, the node blocks the **switchcc** command.

You cannot execute **switchcc** during a configuration-copy. If you attempt it, the system displays the message “Core card redundancy unavailable.”

Cards on Which This Command Runs

PXM45

Syntax

switchcc

Related Commands

None

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

Example

Attempt a switchcc without a standby PXM45.

```
raviraj.7.PXM.a > switchcc
Do you want to proceed (Yes/No)? y

 Core card redundancy unavailable

raviraj.7.PXM.a >
```

telnet

Telnet to another switch

The **telnet** command lets you directly telnet to another switch from the current CLI session. Therefore, you do not have to exit the current CLI session and start a new telnet session to reach another switch. This command requires 2.1 or higher software.

No limit exists on the number of telnet hops you can take from an individual CLI session. However, an individual node has a limit of 15 telnet sessions. Therefore, a limit of 15 users can start telnet sessions.

Cards on Which Command Executes

PXM45

Syntax

```
telnet [-E <escape character>] [-R <traceroute character>] <ip addr>
[[0x|X|x] <tcp port>]
```

Syntax Description

-E	(Optional.) The escape character for terminating the next telnet session in the chain. The purpose of the escape sequence (Esc key then *escape character*) is to terminate the telnet attempt if the destination node is unable to accept the telnet session for reasons such as: the destination switch is down or unreachable; its in backup boot mode; or it detects that TCP communication is down. If you use the escape sequence, the session falls back to the first switch in the telnet chain. The default value for *escape character* is "Q."
-R	(Optional.) The character for triggering a printout of details for all successive hops in the telnet session. The default value for the *traceroute character* is "g."
ip addr	The IP address of the next switch in the chain.
tcp port	(Optional.) The destination *tcp port*. The default TCP port is identical to the destination telnet port. Most applications call for the default. In addition, by default you can enter the number in decimal format, but you can also enter the optional TCP port address in hexadecimal format by preceding the TCP destination with any one of the strings "0x," "X," or "x."

Related Commands

exit, **bye**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Telnet to the switch with IP address 172.39.52.76. The destination switch is of uncertain software level, so check to see whether it can let you telnet to another node. Eventually, return to the source node by exiting the destination node.

```
pop20two.8.PXM.a > telnet 172.39.52.76
Trying 172.39.52.76...

Connected to 172.39.52.76

Login: superuser
password:

8850NY.7.PXM.a > ? telnet
(nothing appropriate)

8850NY.7.PXM.a > dspversion

  Image Type   Shelf Type    Card Type         Version   Built On
  ----------   ----------   ----------    ------------   ------------
     Runtime          MGX        PXM45       2.0(10.2)   Sep 20 2000, 08:40:25
        Boot          MGX        PXM45         2.0(10)   -

8850NY.7.PXM.a > exit

(session ended)

Connection closed by foreign host.
```

timeout

Timeout—specifies the number of seconds of idle time for the current user-session.

The **timeout** command lets you extend the amount of idle time in a user-session from the default of 10 minutes. If you do not specify a timeout period, the system displays the current timeout. At the end of the session, the system logs you out.

To disable the session timeout function, specify 0 seconds.

Note The **timeout** command is the same as the **sesntimeout** command.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
timeout
[time_out]
```

Syntax Description

time_out	Number of idle seconds time allowed for the session.

Related Commands

sesntimeout

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Examples

Display the current timeout.

```
pinnacle.7.PXM.a > timeout
The timeout period for this session is currently 600 second(s)
pinnacle.7.PXM.a >
```

Set the session timeout threshold to 100 minutes (6000 seconds).

```
pinnacle.7.PXM.a > timeout 6000
The timeout period for this session is now set to 5000 second(s)
 pinnacle.7.PXM.a >
```

users

Users—display the ID of each user logged into a card.

The **users** command shows the:

- Access method and port (telnet session to the PXM45, for example)
- Current card slot
- Idle time for the user session (can depend on the **sesntimeout** command)
- User-name (the login name)
- Point from which the user gained access (for example, an IP address in the case of a telnet session or the word “console” if the user logged in through a local terminal at the console port)

Note that **users** shows the current user sessions, whereas the **dspusers** command shows the names of all the user accounts on the switch whether or not a corresponding user has logged in.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
users
```

Syntax Description

This command takes no parameters.

Related Commands

dspusers, **adduser**, **cnfuser**, **timeout**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current users on the switch. Only one user-session is currently running. The user telnetted to the switch from IP address 10.18.247.21. The idle time is 0 because the current user has just executed the **users** command. If other user-sessions were running and one or more were idle, the idle time for each user would be a non-zero number. Change to the AXSM CLI and execute **users**.

```
pop20one.7.PXM.a > users

    Port        Slot      Idle       UserId       From
    -------------------------------------------------------------
    telnet.01 *    7     0:00:00     davids4      10.18.247.21

pop20one.10.AXSM.a > users

    Port        Slot      Idle       UserId       From
    -------------------------------------------------------------
    telnet.01     10     0:00:00     cisco        0.0.0.0
    smterm.03 *   10     0:00:00     davids4      slot 7
```

Equipment and Resource Provisioning

This chapter describes the commands that let you activate, configure, display, and delete resources at various levels of the switch. These levels are:

- Node
- Card
- Line
- Port

The MGX 8850 node uses the concept of a *bay*. The bay refers to the back card position. The T3/E3, OC-3, and OC-12 versions of the ATM Switching Service Module (AXSM) can have two back cards, one in bay 1 (upper slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation Guide*.

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order that you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the **Return** key or **Enter** key.

If you press the **Return** key or **Enter** key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 3-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 switch features the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation Guide*, Release 2.

Table 3-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–4	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 3-2 and Table 3-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 3-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 3-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 3-4.

Table 3-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 3-4 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1

- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

addapsln

Add APS Line

Creates (designates) a pair of lines (*workingIndex, protectIndex*) as APS lines. To configure the APS parameters, use the **cnfapsln** command after creating the lines using the **addapsln** command.

APS Overview

Automatic Protection Switching (APS) is a standards-based redundancy scheme which enhances network reliability by protecting against line failure. APS is defined in Bellcore and ITU standards for North American SONET and international Synchronous Data Hierarchy (SDH) optical network links. The relevant standards are Bellcore GR-253 and ITU-T G.783.

APS enables a pair of SONET lines to be configured for line redundancy. The APS pair consists of a working line (*workingIndex*) and a protection line (*protectIndex*), where one line is active and the other is a backup. Whether or not the backup line passes real traffic while in standby mode depends on the APS architecture mode (*archmode*).

Coordination of line switching is controlled by an in-band signaling protocol. If the fiber optic carrier for the active line is severed or damaged, the in-band signaling protocol must detect the fault within 10 milliseconds. After the in-band signaling protocol has detected the fault, it must switch the user traffic to the standby line within 50 milliseconds.

When the *revertive* option is enabled (see **cnfapsln**), the in-band signaling protocol will attempt to switch the user traffic back to the working line from the protection line after the working line becomes functional again. However, it must wait for the configured time period (*wait to restore*) to elapse.

Direction

APS can be configured in two directions (see *direction* parameter in **cnfapsln**), bidirectional and unidirectional. Bidirectional means that both the receiving and transmitting paths are switched. Unidirectional means that only the affected path, receiving or transmitting, is switched.

Same-card APS

In same-card APS, the working bay and protection bay must be the same, and the working line and protection line must be adjacent.

Architecture mode 1:1 is supported only on same-card APS.

Cross-card APS

In cross-card APS, the working bay and line number, and the protection bay and line number must be the same. The working slot and the protection slot must be adjacent. Card redundancy must be configured on the two cards before cross-card APS can be added.

Architecture modes 1+1, Annex 1+1, and Straight 1+1 Nok1k2 are supported on same-card as well as cross-card APS.

APS Architecture Modes (*archmode*)

The MGX 8850 supports two architecture modes. You must select one of these architecture modes in the *archmode* parameter when you execute the **addapsln** command.

- APS 1:1—provides line redundancy with traffic on the active line only
- APS 1+1—provides line redundancy with traffic on both lines

Note APS 1:1 is only supported on Model B software. It is not supported on Model A software.

Cards on Which This Command Runs

AXSM

AXSM Syntax

addapsln <*workingIndex*> <*protectIndex*> <*archmode*>

AXSM Syntax Description

workingIndex	Slot number, bay number, and line number of the working line in the format: *slot.bay.line*
protectIndex	Slot number, bay number, and line number of the protection line in the format: *slot.bay.line*
archmode	The APS architecture mode to be used on the working/protection line pairs. • 1 = 1+1 – provides line redundancy with traffic on both lines. • 2 = 1:1 – provides line redundancy with traffic on the active line only.

Related Commands

cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: log	State: active	Privilege: GROUP1

Example

The following example assigns 1+1 APS redundancy to two lines on the same card:

```
MGX8850.9.AXSM.a > addapsln 9.2.1 9.2.2 1
```

addcontroller

Add Controller

Identifies a network control protocol to the Virtual Switch Interface (VSI) that runs on the node. If you do not identify a network control protocol (or simply *controller*), the switch does not use it. Adding a controller through the **addcontroller** command requires the following information:

- The type of controller, such as PNNI
- Where the controller runs—internally, on the local PXM45, or externally, through a virtual connection

Note Currently, the only supported controller is PNNI, and it runs only as an internal controller.

Use **addcontroller** before you create resource partitions the service modules. For information on resource partitions, see the description of **addpart** (or **addrscprtn**).

If you discover an error in your controller specification after you add it, delete the controller by executing **delcontroller** then add it again.

Cards on Which This Command Runs

PXM45

Syntax

The syntax differs for internal and external controllers:

Internal controller:

```
addcontroller <cntrlrId> i <cntrlrType> <lslot> cntrlrName]
```

External controller:

```
addcontroller <cntrlrId> x<cntrlrType> <lslot> <bay> <line> <vpi> <vci> [cntrlrName]
```

Syntax Description for Internal Controller

cntrlrId	A number in the range 1–3 that identifies a network controller. The numbers are reserved, as follows: 1 = PAR (Portable AutoRoute)—currently not used 2 = PNNI 3 = LSC (Label Switch Controller, also known as MPLS for Multiprotocol Label Switch Controller)—currently not used **Note** For an internal controller, the controller ID (*cntrlrId)* must be the same as the controller type (*cntrlrType*). Currently, the only available controller is PNNI, and only internal controllers are supported. Therefore, the only valid argument string for **addcontroller** is “2 i 2.” See the example.
i	Keyword indicating that this controller is internal.

cntrlrType	A number in the range 1–3 that identifies a network controller. For internal controllers, the numbers are reserved, as follows: 1 = PAR (Portable AutoRoute)—currently not used 2 = PNNI 3 = LSC (Label Switch Controller, also known as MPLS for Multiprotocol Label Switch Controller)—currently not used **Note** For an internal controller, the controller type (*cntrlrType*) must be the same as the controller ID (*cntrlrId)*.
lslot	The logical slot number on which the controller resides. For the PXM45, *lslot* is 7 regardless of which card is active.
cntrlrName	(Optional) A string to serve as a name for the controller.

Syntax Description for External Controller

Note Currently, the switch does not support an external controller.

cntrlrId	A number in the range 4–20 that identifies a controller. (*cntrlrId* 1–3 is reserved for PAR, PNNI, and MPLS/LSC, respectively.) For external controllers, *cntrlrId* can differ from *cntrlrType.*
x	Keyword indicating that this controller is external.
cntrlrType	A number in the range 1–3 that identifies the controller: 1 = PAR (Portable AutoRoute)—currently not used 2 = PNNI 3 = LSC (Label Switch Controller, also known as MPLS for Multiprotocol Label Switch Controller)—currently not used
lslot	The number of the slot that has the virtual connection through which the controller operates. The ranges are 1–6 and 9–16.
bay	Applies to only external controllers. Upper or lower position of the back card. Type a 1 for upper bay or a 2 for lower bay.
line	Applies to only external controllers. A number specifying the physical line number. The range is 1 through the highest number of lines on the back card. See Table 3-1 for a list.
vpi	Applies to only external controllers. VPI in the range 0–255.

vci	Applies to only external controllers. VCI in the range 1–65535.
cntrlrName	(Optional) A string to serve as a name for the controller.

Related Commands

dspcontrollers, **delcontroller**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Add an internal PNNI controller. Note that, as stated in the Syntax Description for an Internal Controller, the controller ID matches the controller type. The optional controller name is “pnni.” No system response appears unless an error occurs.

```
MGX8850.PXM45.a > addcontroller 2 i 2 7 pnni
```

addfdr

Add Feeder

Adds a feeder node connection to the specified port (*ifNum*). The interface numbers of active ports are displayed in the **dspports** command report. LMI is up by default when you use **addfdr**.

When adding a feeder node, the following conditions apply:

- You can add a feeder node only to an already existing port (*ifNum*).
- You cannot add a feeder node to a port that already has a connection established on it.
- You cannot add a feeder node to a port with ILMI enabled.
- You cannot enable ILMI on a port that has a feeder node connection on it.

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

addfdr <*ifNum*>

Syntax Description

ifNum	The interface number of the port to which the feeder node connection will be added. The interface numbers of active ports are displayed in the **dspports** command report.

Card(s) on Which This Command Executes

AXSM

Related Commands

delfdr, dspfdr, dspfdrs

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.AXSM.a > addfdr 1
```

addpart

Add Resource Partition

Add a resource partition. Before you add resource partitions, be sure you have a plan for future developments, such as the addition of a new controller.

Note The **addpart** and **addrscprtn** commands are identical. The name 'addrscprtn' is consistent with the corresponding command in Release 1 of the MGX 8850 node. Use the command name that suits you. The same identification applies to commands that display and delete a resource partition. In fact, you can add a resource partition with **addrscprtn** then display and delete that partition by executing **dsppart** and **delpart**, respectively.

A resource partition consists of:

- Guaranteed percentage of bandwidth.
- VPI and VCI ranges.
- Guaranteed minimum and maximum number of connections.

Note The maximum number of connections must be greater than 10.

Before adding a resource partition, you must:

- Activate physical lines on the card (**upln** and optional **cnfln**).
- Add logical ports to the physical lines (**addport** or optional **cnfport**).
- Run the **addcontroller** command on the PXM45 to identify the controller type to the Cisco Virtual Switch Interface (VSI) and give that controller an ID number. The **addpart** command requires this controller ID as an argument.

Note For a virtual network-to-network interface (VNNI—also known as a virtual trunk), you can configure one VNNI per port and one port per partition. Specify a VNNI through **addport**.

Cards on Which This Command Runs

AXSM

Syntax

```
addpart <if_num> <part_id> <ctrlr_id> <egrminbw> <egrmaxbw> <ingminbw> <ingmaxbw>
<min_vpi> <max_vpi> <min_vci> <max_vci>
minConns
maxConns
```

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_ID	The partition ID number in the range of 1–5.

ctrlr_id	A number in the range 1–3 that identifies a network controller. The numbers are reserved, as follows: 1 = PAR (Portable AutoRoute)—currently not used 2 = PNNI 3 = LSC (Label Switch Controller, also known as MPLS for Multiprotocol Label Switch Controller)—currently not used
egrminbw	A guaranteed percentage of egress bandwidth. Each unit of *egrminbw* is 0.000001 of the total bandwidth on the port. (An *egrMinBw* of 1000000 = 100%.) This approach provides a high level of granularity.
egrmaxbw	A maximum percentage of the bandwidth. Each unit of *egrmaxbw* is 0.000001 of the total bandwidth available to the port. (An *egrMaxBw* of 1000000 = 100%.) The resulting bandwidth must be at least 50 cps.
ingminbw	A guaranteed percentage of the ingress bandwidth. Each unit of *ingminbw* is 0.000001 of the total bandwidth available to the port. For example, an *ingMinBw* of 1000000 = 100%.
ingmaxbw	A maximum percentage of the ingress bandwidth. Each increment of *ingmaxbw* is 0.000001 of the total bandwidth on the port. For example, an *ingMaxBw* of 1000000 = 100%. Note that the maximum ingress bandwidth must be at least 50 cps.
minvpi	Minimum VPI. For NNI, the range is 0–4095. For UNI, the range is 0–255.
maxvpi	Maximum VPI in the range 0–4095 for an NNI. For a UNI, the range is 0–255. The *maxvpi* cannot be less than the *minvpi*.
minvci	Minimum VCI in the range 0–2000 (OC-48 only) or 32–65535.
maxvci	Maximum VCI in the range 0–2000 (OC-48 only) or 32–65535.
minConns	A guaranteed number of connections. The range is between 0 and the maximum number of connections in the port group. See **dspcd** for information on port groups.
maxConns	A maximum number of connections. The range is between 10 and the maximum number of connections in the port group. See **dspcd** port group information. The value of *maxConns* cannot be less than the value of *minConns*.

Related Commands

cnfpart, **delpart**, **dspparts**, **dsppart**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Create a resource partition with the following parameters:

- Logical port 4 (already created by executing **addport**)
- Partition number 4
- Controller ID 2 (the reserved ID for PNNI)
- 10% of the bandwidth in the egress and ingress directions reserved for this partition
- The range for VPIs is 10–110, the range for VCIs is 100–2000
- Minimum guaranteed connections is 100, maximum number of connections is 500

```
MGX8850.3.AXSM.a > addpart 4 4 2 100000 100000 100000 100000 10 110 100 2000 100 500
```

Check the configuration with **dspparts** and **dsppart**.

```
MGX8850.3.AXSM.a > dspparts
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
----------------------------------------------------------------------------
 4   4    2  100000  100000  100000  100000   10  110   100  2000    100    500

MGX8850.3.AXSM.a > dsppart 4 4
  Interface Number              : 4
  Partition Id                  : 4        Number of SPVC: 0
  Controller Id                 : 2        Number of SPVP: 0
  egr Guaranteed bw(.0001percent): 100000   Number of SVC : 0
  egr Maximum bw(.0001percent)   : 100000
  ing Guaranteed bw(.0001percent): 100000
  ing Maximum bw(.0001percent)   : 100000
  min vpi                       : 10
  max vpi                       : 110
  min vci                       : 100
  max vci                       : 2000
  guaranteed connections        : 100
  maximum connections           : 500
```

addport

Add Port—add a logical port (to the VSI slave).

A logical port is associated with a physical line. For a UNI or NNI, a line can support one logical port. For a virtual NNI (VNNI), a line can support multiple logical ports.

The range of logical port numbers is 1–60 for the AXSM regardless of the interface type (UNI, NNI, or VNNI). For example, if a card supports 4 lines and all lines support UNI, the card can have 4 logical ports whether their numbers are 1–4, 11–14, 57–60, and so on.

You can only execute **addport**, on a currently active line. See **upln**.

Note The maximum number of logical ports for the entire node is 99. This value is imposed by the PXM45/A.

Note If your are going to use card statistics, you must execute **cnfcdstat** before you add any logical ports using **addport**. You cannot configure card statistics once you have any logical ports added.

The information you specify with **addport** consists of the:

- Logical port number
- Line number and bay number
- Guaranteed rate and the maximum rate—currently the same for all interface types
- Service class template ID for the logical interface
- Type of interface (UNI, NNI, or VNNI)
- VPI for all connections on the port if the interface type is VNNI

The node supports a template approach to specifying parameters for large numbers of connections. (You can modify an individual connection as needed using **cnfcon**.) The name of such a template is Service Class Template (SCT). The targets of template application are the logical ports on the one hand and the card itself on the other. The **addport** command lets you specify an SCT for a port, and **cnfcdsct** lets you specify an SCT for the card. You can specify the same or different SCT number for either the port or card-level, but you definitely need to specify an SCT for each card and port. The system automatically assigns a card-level SCT to a card or a port-level SCT of the same number to a port. (For example, if you specify SCT 2 for a port, the system does not assign *card-level* SCT 2 to that port.)

Cisco Systems provides SCT numbers 2 and 3. SCT 2 contains policing parameters, but SCT 3 does not. You should specify SCT 2 or 3 or create new templates by modifying SCT 2 or 3 in the Cisco WAN Manager application and saving them with different SCT numbers. To see the actual values in an SCT, use **dspportsct** for a port SCT or **dspcdsct** for a card-level SCT. To see a list of SCT files on the disk, execute **cd** to get to the SCT directory, then execute **ls** to find the directory named “AXSM.”

Until you specify an SCT, the AXSM has a default SCT of 0. The system uses SCT ID = 0 when:

- The AXSM is powered-up for the first time.
- The card’s database is rebuilt.
- The card is rebooted and the user-specified SCT file for a particular port is corrupt or missing. In this situation, the default applies to only the affected port.

Cards on Which This Command Runs

AXSM

Syntax

```
addport <ifNum> <bay.line> <guaranteedRate> <maxrate> <sctID> <ifType>
[vpi]
```

Note For all ports, *guaranteedRate* must be the same as *maxrate*.

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60.
bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.
guaranteedRate	Guaranteed rate on a port in cells per second. For all interface types (UNI, NNI, or VNNI), *guaranteedRate* must be the same as *maxrate*. The total guaranteed rates cannot exceed the highest value in the following ranges: OC48: 50–5651320 cps OC12: 50–1412830 cps OC3: 50–353207 cps T3: 50–96000 cps for PLCP or 104268 cps for ADM E3: 50–80000 cps
maxRate	Maximum rate on a logical port in cells/second. For all interface types (UNI, NNI, or VNNI), *guaranteedRate* must be the same as *maxrate*. The total maximum rates cannot exceed the highest value in the following ranges: OC48: 50–5651320 cps OC12: 50–1412830 cps OC3: 50–353207 cps T3: 50–96000 cps for PLCP or 104268 cps for ADM E3: 50–80000 cps

sctID	The ID of a service class template (SCT) for the port. The range is 0–255. The SCT file must exist on the PXM45 disk. See **cnfcdsct**. Cisco Systems provides SCT numbers 2 and 3. You can create more SCTs by modifying an existing SCT through the Cisco WAN Manager application and saving it with another SCT number. Subsequently, you can assign the new SCT to the port by the *sctID* parameter in **cnfport**. **Note** Currently, the system does not support certain parameters in the SCTs, so you can specify them through **addcon**, **cnfcon**, or Cisco WAN Manager. These parameters are PCR, SCR, and ICR.
ifType	Specifies the interface type: 1 = UNI 2 = NNI 3 = Virtual path NNI (VNNI)
vpi	Virtual path identifier in the range 1–4095.

Related Commands

cnfport, **delport**, **dspport**, **dspports**, **dspportsct**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Create logical port 1 on line 1, bay 1. Currently, the minimum and maximum cells per second must be the same and are 96000 cps in this example. The egress SCT file ID is 4. The interface type is NNI (2).

```
MGX8850.9.AXSM.a > addport 1 1.1 96000 96000 4 2
```

addred

Add Redundancy

Links two slots to support card-level redundancy for a pair of service modules. The pair consists of a primary slot and a secondary slot. Currently, the only form of redundancy for service modules is 1:1. Although 1:*n* appears in the Help display, it is reserved for future use.

To see the slots where redundancy exists, use **dspcd** or **dspcds**.

Cards on Which This Command Runs

PXM45

Syntax

```
addred <redPrimarySlotNum> <redSecondarySlotNum> <RedType>
```

Syntax Description

redPrimarySlotNum	Slot number that contains the primary card of the pair. The ranges are 1–6 and 9–14 on the MGX 8850 switch.
redSecondarySlotNum	Slot number that contains the secondary card. The ranges are 1–6 and 9–14 on the MGX 8850 switch.
RedType	Value to set type of redundancy to be deployed on the PXM45. 1 = 1:1 Y Cable 2 = 1:*N* (reserved for future use)

Related Commands

dspred, delred, switchredcd

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example adds 1:1 Y Cable redundancy in slots 7 and 8.

```
MGX8850.7.PXM.a > addred 7.8.1
```

addrscprtn

Add Resource Partition

Add a logical partition of resources for a network controller on a port. Before you add resource partitions, be sure a plan exists for future developments, such as the addition of a new controller.

Note The **addpart** and **addrscprtn** commands are identical. The name ‘addrscprtn’ is consistent with the corresponding command in Release 1 of the MGX 8850 switch. Use whichever command name suits you purpose. The same identification applies to commands that display and delete a resource partition. In fact, you could add a resource partition with **addrscprtn** then display and delete that partition by executing **dsppart** and **delpart**, respectively.

A resource partition consists of:

- Guaranteed percentage of bandwidth.
- VPI and VCI ranges.
- Guaranteed minimum and maximum number of connections.

Note The maximum number of connections must be greater than 10.

Before adding a resource partition, you must:

- Activate physical lines on the card (**upln** and optional **cnfln**).
- Add logical ports to the physical lines (**addport** and optional **cnfport**).
- Execute **addcontroller** on the PXM45 to identify a *controller type* to the Cisco Virtual Switch Interface (VSI) and give that controller an ID number. The **addrscprtn** command takes this controller ID as an argument.

The current network control application is PNNI. For possible future use, plan the partitioning to reflect possible use of MPLS or other controllers.

Note For VNNIs (virtual trunks), you can configure one VNNI per port and one port per partition. Specify the VNNI interface type through the **addport** command.

Cards on Which This Command Runs

AXSM

Syntax

```
addrscprtn <if_num> <part_id> <ctrlr_id> <egrminbw> <egrmaxbw> <ingminbw> <ingmaxbw>
<minVpi> <maxVpi> <minVci> <maxVci>
minConns
maxConns
```

Note The maximum number of connections must be at least 10.

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_ID	The number of the partition in the range 1–5.
ctrlr_id	The number of the controller in the range 1–20.
egrminbw	A guaranteed percentage of egress bandwidth. Each unit of *egrMinBw* is 0.00001 of the total bandwidth on the port. (An *egrMinBw* of 1000000 = 100%.) This approach provides a high level of granularity.
egrmaxbw	A maximum percentage of the bandwidth. Each unit of *egrMaxBw* is 0.00001 of the total bandwidth available to the port. (An *egrMaxBw* of 1000000 = 100%.) The resulting bandwidth must be at least 50 cps.
ingminbw	A guaranteed percentage of the ingress bandwidth. Each unit of *ingMinBw* is 0.00001 of the total bandwidth available on a port. For example, an *ingMinBw* of 1000000 = 100%.
ingmaxbw	A maximum percentage of the ingress bandwidth. Each increment of *ingMaxBw* is 0.00001 of the total bandwidth on the port. For example, an *ingMaxBw* of 1000000 = 100%. Note that the maximum ingress bandwidth must be at least 50 cps.
minVpi	Minimum VPI. For NNI, the range is 0–4095. For UNI, the range is 0–255. For a virtual trunk (VNNI interface type in the **addport** command), the *minVpi* must be the same as the *maxVpi*.
maxVpi	Maximum VPI in the range 0–4095 for an NNI. For a UNI, the range is 0–255. The *minVpi* cannot be less than the *maxVpi*.
minVci	Minimum VCI in the range 0–2000 (OC-48 only) or 32–65535.
maxVci	Maximum VCI in the range 0–2000 (OC-48 only) or 32–65535.
minConns	A guaranteed number of connections. The range is between 0 and the maximum number of connections in the port group. See **dspcd** for information about port groups.
maxConns	A maximum number of connections. The range is between 0 and the maximum number of connections in the port group. See **dspcd** port group information. *maxConns* cannot be less than *minConns*.

Related Commands

cnfrscprtn, **delrscprtn**, **dsprscprtns**, **dsprscprtn**

Attributes

Log: log	State: active	Privilege: GROUP1

clrbecnt

Clear Bit Error Count

The **clrbecnt** command lets you clear the APS-related bit error counters for a working line. To see the contents of the error counters, use the **dspbecnt** command.

Cards on Which This Command Runs

AXSM

Syntax

`clrbecnt <working-bay.line>`

Syntax Description

working-bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, dspbecnt

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

clrfdrstat

Clear Feeder Statistics

Clears the LMI and node statistics for the feeder on the specified port (*ifNum*).

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

```
clrfdrstat <ifNum>
```

Syntax Description

ifNum	The interface number of the port on which to clear the feeder statistics. The interface numbers of active ports are displayed in the **dspports** command report.

Card(s) on Which This Command Executes

AXSM

Related Commands

dspfdrstat

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

```
clrfdrstat 1
```

cnfapsln

Configure APS Line

Configures the APS parameters for a line (*working line*). Use the **cnfapsln** command after creating the line using the **addapsln** command.

See the description for the **addapsln** command for a detailed explanation of Automatic Protection Switching (APS).

Cards on Which This Command Runs

AXSM

Syntax

```
cnfapsln
-w <working line>
[-sf  SignalFaultBER ]
[-sd SignalDegradeBER]
[-wtr Wait To Restore]
[-dr direction]
[-rv revertive]
```

Syntax Description

workingLine	Slot number, bay number, and line number of the active line to configure, in the format: *slot.bay.line* `Example: -w 1.1.1`
SignalFaultBER	A number between 3 and 5 indicating the Signal Fault Bit Error Rate (BER), in powers of ten: • $3 = 10^{-3}$ • $4 = 10^{-4}$ • $5 = 10^{-5}$ `Example: -sf 3`
SignalDegradeBER	A power if 10 in the range 5–9 that indicates the Signal Degrade Bit Error Rate (BER): • $5 = 10^{-5}$ • $6 = 10^{-6}$ • $7 = 10^{-7}$ • $8 = 10^{-8}$ • $9 = 10^{-9}$ `Example: -sd 5`

Wait To Restore	The number of minutes to wait after the working line has become functional again, before switching back to the working line from the protection line. The range is 5–12. `Example: -wtr 5`
direction	1: unidirectional, 2: bidirectional `Example: -dr 2` Bidirectional means that both the receiving and transmitting paths are switched. Unidirectional means that only the affected path, receiving or transmitting, is switched.
revertive	1: non-revertive, 2: revertive `Example: -rv 1`

Related Commands

addapsln, delapsln, dspapsln, dspapslns, switchapsln, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
cnfapsln -w 1.1.1 -sf 3 -sd 5 -wtr 5 -dr 2 -rv 1
```

cnfatmln

Configure ATM Line

Configures the ATM layer cell header for the specified line (*bay.line*).

You must configure the ATM layer cell header for a line before you activate the line using **upln**, or before you add a logical port to the line using **addport**.

Cards on Which This Command Runs

AXSM

Syntax

```
cnfatmln
-ln <bay.line>
-sps <PayloadScramble>
-nch <cellhdr>
-ncp <cellpayload>
-hcs <hcs>
```

Syntax Description

-ln	Specifies the bay and line number in the format *bay.line*. The *bay* is 1 or 2. The *line* can be from 1 to the highest numbered line on the back card. For the line number ranges on an AXSM model, see Table 3-1.
-sps	Specifies the number that enables (1) or disables (2) payload scrambling. The default value for *PayloadScramble* is enabled. The setting must be the same at both ends of the line and throughout the path.
-nch	Specifies the four-byte hexadecimal number to serve as the null cell header (*cellhdr*). The range for *cellhdr* is all 0s through ffffffff.
-ncp	Specifies a 8-bit hexadecimal byte to serve as the null cell header. The range for *cellpayload* is 1–ff. The default is 6a.
-hcs	Specifies the number to disable (1) or enable (2) HCS coset The default is enabled.

Related Commands

dspatmln

Attributes

Log: log	State: active	Privilege: GROUP1

Example

For AXSM, line 1, bay 1, disable payload scrambling and specify a null cell header.

```
MGX8850.7.AXSM.a > cnfatmln -ln 1.1 -sps 2 -nch ab12abab
```

For AXSM, line 1, bay 1, enable payload scrambling and specify null cell headers.

```
MGX8850.1.AXSM.a > cnfatmln -ln 1.1 -sps 1 -nch 1a1a1a1a -ncp aa
```

cnfcdsct

Configure Card SCT

Assign a service class template (SCT) to an AXSM at the card level. The template contains bandwidth and policing parameters for an AXSM.

Note Currently, the system does not support certain parameters in the service class templates (SCTs), so you can specify them through **addcon**, **cnfcon**, or Cisco WAN Manager.
These parameters are (when applicable) PCR, SCR, and ICR.

Usage Guidelines

The **cnfcdsct** command is card level because it applies to the card's interface to the backplane. (See **addport** for specifying an SCT for a port.) The following characteristics apply to **cnfcdsct**.

- A valid SCT file must exist on the PXM45 disk before you execute **cnfcdsct**. To see a list of SCT files on the disk, execute **cd** to get to the SCT directory, then execute **ls** to see the directory named AXSM.
- You must execute **cnfcdsct** only when the card is down.
- You cannot change the SCT configuration if any ports, lines, or partitions are configured.
- To see the ID of the current SCT, use **dspcd** for the card-level SCT or **dspport** for a port-level SCT.
- To see the actual contents of SCT 2 and SCT 3, use **dspcdsct**.

Background

The node supports a template approach to specifying parameters for large numbers of connections. (You can modify an individual connection as needed through **cnfcon**.) The name of such a template is *service class template* (SCT). The targets of template application are the logical ports on the one hand and the card itself on the other. The **addport** command lets you specify an SCT for a port, and **cnfcdsct** lets you specify an SCT for the card. You can specify the same or different SCT number for either the port or card-level, but you definitely need to specify an SCT for each card and port. The system automatically assigns a card-level SCT to a card or a port-level SCT of the same number to a port. (For example, if you specify SCT 2 for a port, the system does not assign *card-level* SCT 2 to that port.)

Cisco Systems provides SCT numbers 2 and 3. SCT 2 contains policing parameters, but SCT 3 does not. You should specify SCT 2 or 3 or create new templates by modifying SCT 2 or 3 in the Cisco WAN Manager application and saving them with different SCT numbers in the range 1–255.

Until you specify an SCT, the AXSM has a default SCT of 0. The system uses SCT 0 when:

- The AXSM is powered-up for the first time.
- The card's database is rebuilt.
- The card is rebooted and the user-specified SCT file for a particular port is corrupt or missing. In this situation, the default applies to only the affected port.

Cards on Which This Command Runs

AXSM

Syntax

```
cnfcdsct <SCT-id>
```

Syntax Description

SCT-id	Number of the SCT at the card-level. The range is 1–255.

Related Commands

dspcdsct, **dspcd**, **dspsct**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Specify card-level SCT number 3 for the current AXSM.

```
MGX8850.9.AXSM.a > cnfcdsct 3
```

The SCT file must reside on the PXM disk before you use this command, or it fails and displays the error message in the following example:

```
MGX8850.1.6.AXSM.a > cnfcdsct -sct 5
ERR:SCT file not present. Use tftp to load it on PXM disk
Set failed due to illegal option value(s)

Syntax: cnfcdsct "<sctID>"
        sctID -- SCT file id between 0 and 255
```

cnfln

Configure Line

Configures a line on the current service module. Use **cnfln** after you have activated the line using **upln**.

Note You cannot configure a line that currently has any configured virtual interfaces on it.

Cards on Which This Command Runs

AXSM

Note The syntax varies according to the line type, so each line type has a description.

Syntax for SONET Line

cnfln
-**sonet** <*bay.line*>
-**slt** <*LineType*>
-**clk** <*clockSource*>

Syntax Description (SONET)

All parameters are keyword driven. Therefore, the order that you enter them does not matter.

-sonet	Identifies the bay (1 or 2) and the number of the line in the format *bay.line*. The range for *line* is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.
-slt	A number that identifies the type of SONET line. The range for *LineType* is 1–6 and have the following significance: 1 = sonetSts3c 2 = sonetStm1 3 = sonetSts12c 4 = sonetStm4 5 = sonetSts48c 6 = sonetSTM16
-clk	Determines whether the transmit clock comes from the backplane (local timing) or the receive clock on the line (looped timing). The value for *clockSource* can be 1 or 2: 1 = loop timing: the receive clock is re-directed on the back card to become the transmit clock. 2 = local timing: (default) clock from the backplane is the transmit clock.

Syntax for T3/E3 Line

```
cnfln -ds3 <bay.line>
-lt <LineType>
-len <LineLength>
-oof <LineOOFCriteria>
-cb <LineAIScBitsCheck>
-rfeac <LineRcvFEACValidation>
-clk <clkSource>
```

Syntax Description (T3/E3)

All parameters are keyword driven. Therefore, the order that you enter them does not matter.

-ds3 Identifies the bay (1 or 2) and the number of the line. The format for *LineNum* is *bay.line*. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

-lt Specifies the type of T3 line. The possible value for a T3 *LineType* is 1 or 2:

- 1 = ds3cbitadm
- 2 = ds3cbitplcp

-len The length of the line in meters. The range for *LineLength* is 1–64000.

-oof A number that specifies the threshold for triggering an Out Of Frame condition. The possible value for *LineOOFCriteria* is 1 or 2 and has the following significance:

- 1 = 3 out of 8
  An Out Of Frame condition is declared if at least 3 of 8 framing bits are in error.
- 2 = 3 out of 16
  An Out Of Frame condition is declared if at least 3 of 16 framing bits are in error.

-cb A number that determines whether the node checks the C-bit status in response to AIS status. The value for *LineAIScBitsCheck* can be 1 or 2 and has the following significance:

- 1 = check the C-bit
- 2 = ignore the C-bit

-rfeac Value to set FEAC (far-end alarm and control) code validation criteria. The value for *LineRcvFEACValidation* can be 1 or 2 and has the following significance:

- 1 = 4 out of 5. A valid FEAC code is declared if 4 of 5 codes match.
- 2 = 8 out of 10

A valid FEAC code is declared when 8 of 10 codes match.

-clk A number that determines whether the transmit clock comes from the backplane ("local timing") or the receive clock on the line ("looped timing"). The value for *clockSource* can be 1 or 2 and has the following significance:

1 = loop timing: receive clock is re-directed on the back card to be the transmit clock.

2 = local timing: (default) clock from the backplane is the transmit clock.

Related Commands

dsplns, **dspln, dnln**, **upln**, **addlnloop**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Configure T3 line 4 on the current card to have B8ZS coding and a length of 10.

```
MGX8850.1.4.AXSM.a > cnfln -ds3 4 2 10
```

Enable frame scrambling for SONET line 1 of the card in bay 1.

```
MGX8850.1.4.AXSM.a > cnfln -sonet 1.1 -sfs 2
```

cnfpart

Configure Resource Partition

Modifies a resource partition. A resource partition on an AXSM consists of minimum and maximum percentages of bandwidth, a VPI/VCI range, and a minimum and maximum number of connections available to a network control application. The current network controller is PNNI. Refer to the description of **addpart** for information on resource partitions.

Note The **cnfpart** and **cnfrscprtn** commands are identical. The name ‘cnfrscprtn’ is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

```
cnfpart
-if <if>
-id <partionID>
-ctlr <controllerID>
-emin <egrMinBw>
-emax <egrMaxBw>
-imin <ingMinBw>
-imax <ingMaxBw>
-vpmin <minVpi>
-vpmax <maxVpi>
-vcmin <minVci>
-vcmax <maxVci>
-mincon <min connections>
-maxcon <max connections>
```

Note The maximum number of connections must be greater than 10.

Syntax Description

Note On a virtual trunk, the *min_vpi* and *max_vpi* must be the same.

-if	Logical interface (port) number. For AXSM, the range is 1–60. This parameter is the same as the *ifNum* that appears in other commands.
-id	Specifies the number of the partition in the range 1–5.
-ctlr	Specifies the number of the controller in the range 1–20.
-emin	Specifies the guaranteed percentage of egress bandwidth. Each unit of *egrMinBw* is 0.00001 of the total bandwidth on the port. (An *egrMinBw* of 1000000 = 100%.) This approach provides a high level of granularity.

-emax	Specifies the maximum percentage of the bandwidth. Each unit of *egrMaxBw* is 0.00001 of the total bandwidth available to the port. (An *egrMaxBw* of 1000000 = 100%.) The resulting bandwidth must be at least 50 cps.
-imin	Specifies the guaranteed percentage of the ingress bandwidth. Each unit of *ingMinBw* is 0.00001 of the total bandwidth available to the port. For example, an *ingMinBw* of 1000000 = 100%.
-imax	Specifies the maximum percentage of the ingress bandwidth. Each increment of *ingMaxBw* is 0.00001 of the total bandwidth on the port. For example, an *ingMaxBw* of 1000000 = 100%. Note that the maximum ingress bandwidth must be at least 50 cps.
-vpmin	Specifies the minimum VPI. For NNI, the range is 0–4095. For UNI, the range is 0–255.
-imax	Specifies the maximum VPI in the range 0–4095 for an NNI. For a UNI, the range is 0–255. The *maxvpi* cannot be less than the *minvpi*.
-vcmin	Specifies the minimum VCI in the range 0–2000 (OC-48 only) or 32–65535.
-vcmax	Specifies the maximum VCI in the range 0–2000 (OC-48 only) or 32–65535.
-mincon	Specifies the guaranteed number of connections. The range is between 0 and the maximum number of connections in the port group. See **dspcd** for information about port groups.
-maxcon	Specifies the maximum number of connections. The range is between 10 and the maximum number of connections in the port group. See **dspcd** port group information. *maxConns* cannot be less than *minConns*.

Related Commands

addpart, **delpart**, **dsppart**, **dsppart**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

For logical port 1 on partition number 2, configure the PNNI controller (the number 2) to have 10%–15% of the bandwidth on both the ingress and egress, a VPI range of 20–100, a VCI range of 1–32767, a minimum guaranteed number of 1000 connections (or channels) and a maximum of 2000 connections.

```
MGX-01.1.2.PXM.a > cnfpart 1 2 2 100000 150000 10000 15000 20 100 1 32767 1000 2000
```

cnfport

Configure Port

Configures a logical port on a service module. The system does not display a confirmation upon successful execution, so use **dspport** to check the changes.

Note You cannot use **cnfport** to change the guaranteed rate and maximum rate parameters if a resource partition has been configured for the interface.

You can change the SCT ID if you first down the port by executing **dnport**, then executing **cnfport**. After you change the SCT ID, execute **upport** to return the port to operation. Descriptions of **dnport** and **upport** appear in the chapter, "Troubleshooting Commands."

Cards on Which This Command Runs

AXSM

Syntax

```
cnfport
-if <ifNum>
[-min <guaranteedRate>]
[-max <maxrate>]
[-sct <sctID>]
```

Syntax Description

Note that this command uses the keyword (or command delineator) to identify the parameter that follows it. After you identify the logical port with the *ifNum* parameter, each remaining parameter is optional.

-if	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60.
-min	Specifies the guaranteed rate on a logical port in cells/second. The cumulative guaranteed rate cannot exceed the highest value in the following ranges: OC48: 50–5651320 cps OC12: 50–1412830 cps OC3: 50–353207 cps T3: 50–96000 (PLCP) or 104268 (ADM) cps E3: 50–80000 cps

-max	Specifies the maximum rate on a logical port in cells per second (cps). OC48: 50–5651320 cps OC12: 50–1412830 cps OC3: 50–353207 cps T3: 50–96000 (PLCP) or 104268 (ADM) cps E3: 50–80000 cps
-sct	Specifies the number of a service class template (SCT) for the egress direction. The range is 0–255. Cisco provides a default of SCT 0 as well as SCT numbers 2 and 3. You can modify one of these SCTs through the Cisco WAN Manager application and assign another number within the SCT range to the new SCT. Subsequently, you can assign the new SCT to the port with the *sctID* parameter in **cnfport**. To see the ID of the current SCT for this port, use **dspport**.

Related Commands

addport, **delport**, **dspport**, **dspports**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

For logical port 1, configure a guaranteed minimum of 10000 cps and a maximum rate of 20000 cps.

```
MGX8850.1.6.AXSM.a > cnfport -if 1 -min 10000 max 20000
```

cnfrscprtn

Configure Resource Partition

Modifies a resource partition. A resource partition on an AXSM consists of minimum and maximum percentages of bandwidth, a VPI/VCI range, and a minimum and maximum number of connections available to a network control application. The current network controller is PNNI. Refer to the description of **addrscprtn** for information on resource partitions.

Note The **cnfpart** and **cnfrscprtn** commands are identical. The name 'cnfrscprtn' is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

```
cnfrscprtn
-if <if>
-id <partionID>
-ctlr <controllerID>
-emin <egrMinBw>
-emax <egrMaxBw>
-imin <ingMinBw>
-imax <ingMaxBw>
-vpmin <minVpi>
-vpmax <maxVpi>
-vcmin <minVci>
-vcmax <maxVci>
-mincon <min connections>
-maxcon <max connections>
```

Note The maximum number of connections must be greater than 10.

Syntax Description

Note On a virtual trunk, the *min_vpi* and *max_vpi* must be the same.

-if	Logical interface (port) number. For AXSM, the range is 1–60. This parameter is the same as the *ifNum* appearing in other commands.
-id	The number of the partition in the range 1–5.
-ctlr	The number of the controller in the range 1–20.
-emin	A guaranteed percentage of egress bandwidth. Each unit of *egrMinBw* is 0.00001 of the total bandwidth on the port. (An *egrMinBw* of 1000000 = 100%.) This approach provides a high level of granularity.

-emax	A maximum percentage of the bandwidth. Each unit of *egrMaxBw* is 0.00001 of the total bandwidth available to the port. (An *egrMaxBw* of 1000000 = 100%.) The resulting bandwidth must be at least 50 cps.
-imin	A guaranteed percentage of the ingress bandwidth. Each unit of *ingMinBw* is 0.00001 of the total bandwidth available to the port. For example, an *ingMinBw* of 1000000 = 100%.
-imax	A maximum percentage of the ingress bandwidth. Each increment of *ingMaxBw* is 0.00001 of the total bandwidth on the port. For example, an *ingMaxBw* of 1000000 = 100%. Note that the maximum ingress bandwidth must be at least 50 cps.
-vpmin	Minimum VPI. For NNI, the range is 0–4095. For UNI, the range is 0–255.
-vpmax	Maximum VPI in the range 0–4095 for an NNI. For a UNI, the range is 0–255. The *maxvpi* cannot be less than the *minvpi*.
-vcmin	Minimum VCI in the range 0–2000 (OC-48 only) or 32–65535.
-vcmax	Maximum VCI in the range 0–2000 (OC-48 only) or 32–65535.
-mincon	A guaranteed number of connections. The range is between 0 and the maximum number of connections in the port group. See **dspcd** for information on port groups.
-maxcon	A maximum number of connections. The range is from 10 to the maximum number of connections in the port group. See **dspcd** port group information. *maxConns* cannot be less than *minConns*.

Related Commands

addrscprtn, **delrscprtn**, **dsprscprtns**, **dsprscprtn**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

For logical port 1 on partition number 2, configure the PNNI controller (the number 2) to have 10%–15% of the bandwidth on both the ingress and egress, a VPI range of 20–100, a VCI range of 1–32767, a minimum guaranteed number of 1000 connections (channels) and a maximum of 2000 connections.

```
MGX8850.1.2.AXSM.a > cnfrscprtn 1 2 2 100000 150000 10000 15000 20 100 1 32767 1000 2000
```

delapsln

Delete APS Line

Removes the specified APS line from the switch.

See the description for the **addapsln** command for a detailed explanation of Automatic Protection Switching (APS).

Cards on Which This Command Runs

AXSM

Syntax

```
delapsln
<workingline>
```

Syntax Description

workingline	Slot number, bay number, and line number of the active line to delete, in the format: *slot.bay.line.* `Example: 1.1.1`

Related Commands

addapsln, cnfapsln, dspapsln, switchapsln, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
delapsln 1.1.1
```

delcontroller

Delete Controller

Delete a controller. The **delcontroller** command does not erase the controller software but directs the switch not to use it.

Cards on Which This Command Runs

PXM45

Syntax

```
delcontroller <cntrlrId>
```

Syntax Description

cntrlrId	The controller ID (*cntrlrId*) has a range of 1–20 and is assigned by using the **addcontroller** command. To see all controllers on the switch, use **dspcontrollers**.

Related Commands

addcontroller, **dspcontrollers**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Delete controller 3. In this example, the 3 is the reserved controller ID for LSC.

```
MGX8850.8.PXM.a > delcontroller 3
```

delfdr

Delete Feeder

Deletes a feeder node connection from the specified port (*ifNum*). The interface numbers of active ports are displayed in the **dspports** command report.

Note You cannot delete a port that has feeder node connections on it.

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

delfdr <*ifNum*>

Syntax Description

ifNum	The interface number of the port from which the feeder node connection will be deleted. The interface numbers of active ports are displayed in the **dspports** command output.

Card(s) on Which This Command Executes

AXSM

Related Commands

addfdr, dspfdr, dspfdrs

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
delfdr 8
```

delpart

Delete Resource Partition

Delete a resource partition. Note that you must delete all connections in the resource partition before you can delete it. For information on resource partitions, refer to the description of **addpart**.

Note The **delpart** and **delrscprtn** commands are identical. The name ‘delrscprtn’ is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

`delpart` `<if_num> <part_id>`

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_id	The partition ID number in the range 1–5. Use **dspparts** (or **dsprscprtns**) to see all resource partitions if necessary.

Related Commands

addpart, **cnfpart**, **dsppart**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.9.AXSM.a > delpart 1 1
```

delport

Delete Port

Remove a logical port from a service module. You must delete all connections and resource partitions on the port before you can delete it.

Cards on Which This Command Runs

AXSM

Syntax

`delport` <*ifNum*>

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60.

Related Commands

addport, **cnfport**, **dspport**, **dspports**

Attributes

Log: log	State: active	Privilege: GROUP1

delred

Delete Redundancy

Deletes 1:1 redundancy for a pair of card slots.

Note 1:N redundancy requires a Service Resource Module (SRM) in the switch. The current release does not support SRMs.

Cards on Which This Command Runs

PXM45

Syntax

`delred` *`<redPrimarySlotNumber>`*

Syntax Description

redPrimarySlotNumber	Slot number that contains the primary card of the pair. The ranges are 1–6 and 9–14 on the MGX 8850 switch.

Related Commands

addred, dspred, switchredcd

Attributes

Log: log	State: active	Privilege: SUPER_GP

delrscprtn

Delete Resource Partition

Delete a resource partition. Note that you must delete all connections in the resource partition before you delete it. For information on resource partitions, refer to the description of **addrscprtn**.

Note The **delpart** and **delrscprtn** commands are identical. The name ‘cnfrscprtn’ is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

```
delrscprtn <if_num> <part_id>
```

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_id	The partition ID number in the range 1–5. Use **dsprscprtns** to see all resource partitions if necessary.

Related Commands

addrscprtn, **dsprscprtns**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.9.AXSM.a > delrscprtn 1 1
```

dnlmi

Down Local Management Interface

De-activates the Local Management Interface (LMI) on the specified logical port (*ifNum*).

Cards on Which This Command Runs

AXSM

Syntax

`dnlmi` `<ifNum>`

Syntax Description

ifNum	The interface number of the logical port on which to de-activate the LMI.

Related Commands

uplmi

Attributes

Log: log	State: active	Privilege: ANYUSER

Example

```
dnlmi 2
```

dnln

Down Line

The **dnln** command lets you de-activate a line on the current card. Before you can de-activate a line, you must take the following steps:

Step 1 Remove connections. Use **delcon** or **delcons**.

Step 2 Remove any resource partitions. Use **dsprscprtn** to see existing partitions and **delrscprtn** to remove partitions.

Step 3 Remove all logical ports. Use **dspports** to see existing logical ports on the line and **delport** to remove logical ports.

Cards on Which This Command Runs

AXSM

Syntax

```
dnln <bay.line>
```

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

dspln, **dsplns**, **cnfln**, **upln**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

De-activate line 1 in bay 1.

```
MGX8850.1.AXSM.a > dnln 1.1
```

dnport

Down Port

The **dnport** command disables (or downs) a logical port and thereby halts all traffic on the logical port. The usual purpose for using **dnport** is troubleshooting. The configuration for the port remains intact whether the logical port is a UNI or an NNI. The command for enabling a downed port is **upport**.

For an NNI, the PXM45 de-routes the failed connections then re-routes them through other trunks. After you re-enable an NNI port through **upport**, you cannot return the re-routed connections to the upped port. The PXM45 routes connections over the trunk as needed.

On a UNI, the connections continue to exist, but remain in the failed state until you enable the port by executing **upport**.

Cards on Which This Command Runs

AXSM

Syntax

`dnport` `<`*`ifNum`*`>`

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60. Use **dspports** or **dspport** as needed to determine the need to disable a port.

Related Commands

dspport, **dspports**, **upport**

Attributes

Log: no log	State: active	Privilege: GROUP1

Example

Disable port 1 on the current card.

```
MGX8850.1.AXSM.a > dnport 1
```

dspapsbkplane

Display APS Backplane

Displays whether or not the APS mini-backplane is properly seated with the backcards.

When successful, this command displays:

```
BackPlane:ENGAGED
```

When not successful, this command displays:

```
BackPlane:NOT ENGAGED
```

See the **addapsln** command for an explanation of Automatic Protection Switching (APS).

See the *Cisco MGX 8850 Routing Switch Hardware Installation Guide*, Release 2, for information on installing the APS assembly to the backplane.

Cards on Which This Command Runs

AXSM

Syntax

dspapsbkplane

Syntax Description

No parameters.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, clrbecnt, dspbecnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Example of executing **dspapsbkplane** when an APS assembly is successfully installed to the backplane.

```
MGX 8850.6.AXSM.a > dspapsbkplane 6.1.3
BackPlane:ENGAGED
```

Example of executing **dspapsbkplane** when an APS assembly is not successfully installed to the backplane.

```
MGX 8850.3.AXSM.a > dspapsbkplane 3.1.1
BackPlane:NOT ENGAGED
```

dspapsln

Display APS Line

Displays the configuration of an APS line. This command can be executed for either a working line or a protection line.

See the **addapsln** command for an explanation of Automatic Protection Switching (APS).

Cards on Which This Command Runs

AXSM

Syntax

`dspapsln` *`<working-slot.bay.line>`*

Syntax Description

working-slot.bay.line	Identity of the working line with the format *slot.bay.line*.

Related Commands

addapsln, cnfapsln, delapsln, dspapslns, switchapsln, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the APS configuration for slot 13, bay 1, line 1.

```
MGX8850.13.AXSM.a > dspapsln 13.1.1
Working Index : 13.1.1 Protection Index : 14.1.1
Provisioned Arch : 1+1 Provisioned Direction : bi
Operational Arch : 1+1 Operational Direction : bi
Active Line : working WTR(min) : 5
SFBer 10^-n : 3 SDBer 10^-n : 5
Revertive : No Last User Switch Req : ForcedW->P
Bridge State : WChan Bridged Selector State : Selector Released
Protection Line Pending Request : None
Working Line Pending Request : ForcedSwitch
APS Trouble Mask : ChannelMismatch
Bit Map Req Field Chan Field
Transmit K1 0xe1 Forced Switch Working Channel 1
Receive K1 0x21 Reverse Request Working Channel 1
Current Request 0xe1 Forced Switch Working Channel 1
Bit Map Chan Field Arch Field Dir Mode Field
Transmit K2 0x15 Working Channel 1 1+1 BI
Receive K2 0x25 INVALID 1+1 BI
Alarm State Clear
```

dspapslns

Display APS Lines

Displays all working and protection APS lines on a card. This command can be executed only on an active card. After identifying a particular APS line, you can use **dspapsln** to view details about the line.

See the **addapsln** command for an explanation of Automatic Protection Switching (APS).

Cards on Which This Command Runs

AXSM

Syntax

dspapslns

Syntax Description

This command takes no parameters.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, switchapsln, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display all lines with an APS configuration on the current AXSM.

```
MGX8850.3.AXSM.a > dspapslns
Working Prot.  Conf    Oper    Active  SFBer SDBer WTR   Revt Dir LastUser
Index   Index  Arch    Arch    Line    10^-n 10^-n (min)          SwitchReq
------- -----  ----    -----   ------  ----- ----- ----- ---- --- ----------
  3.1.1  3.1.2  1+1     1+1    working     3     5     5   No uni No Request
  3.1.7  3.1.8  1+1     1+1    working     3     5     5   No uni No Request
```

dspatmln

Display ATM Line

Displays the cell header configuration for the line that was set using **cnfatmln**. The display indicates NNI or UNI cell headers.

Cards on Which This Command Runs

AXSM

Syntax

dspatmln <*bay.line*>

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

cnfatmln, clratmlncnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the line configuration for line 1 of the AXSM-1-2448.

```
MGX8850.1.AXSM.a > dspatmln 1.1

 line  HCScoset PayloadScramble NullCellHdr NullCellPayload
----------------------------------------------------------------------
   1.1   Enable          Enable    1a1a1a1a              aa
```

dspbecnt

Display Bit Error Count

The **dspbecnt** command lets you display the APS-related bit error counters.

Cards on Which This Command Runs

AXSM

Syntax

`dspbecnt` `<working-bay.line>`

Syntax Description

working-bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, dspapsbkplane, dspbecnt

Attributes

Log: no log	State: active	Privilege: ANYUSER

dspcd

Display Card

Display the following information about a card:

- Hardware serial number.
- Firmware revision level. (See the **loadrev** description for an explanation of how to interpret the revision field.)
- Status, possibly including the reason for the last reset (FunctionModuleResetReason) and state of the integrated alarm (cardIntegratedAlarm).
- For a service module only: a count of configured lines, ports, and connections.

Note The connection count includes control VCs when you execute **dspcd** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

- For a service module only: which physical lines constitute a port group and the maximum number of connections in that port group. A port group consists of one to many physical lines. This maximum connection count is a function of the hardware interface type (OC-3, OC-12, and so on). The port group information also shows the number of existing SVCs, SPVCs, and SPVPs.

  Use the maximum number of supported connections to help you configure resource partitions. If a particular resource partition has close to the maximum supported by hardware on a line, few or no connections would be possible in another partition on the same line.

Some of the information that **dspcd** shows can also be displayed using the **version** command, but **version** shows the boot code version in bold.

The total number of connections in the display includes control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that do not include control VCs, use **dsppnport**.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

dspcd

Syntax Description

This command does not take parameters.

Related Commands

dspcds, **version**

PXM45 Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display card details for the current PXM45.

Note The "A1" at the end of the primary software revision and boot firmware revision numbers shows that these versions are pre-release. Refer to the **setrev** description for details.

```
MGX8850.7.PXM.a > dspcd
MGX8850                          System Rev: 02.00   Aug. 02, 2000 23:39:06 GMT
MGX8850                                               Node Alarm: CRITICAL
Slot Number    7    Redundant Slot:  8

                    Front Card          Upper Card          Lower Card
                    ----------          ----------          ----------

Inserted Card:      PXM45               UI Stratum3         PXM HardDiskDrive
Reserved Card:      PXM45               UI Stratum3         PXM HardDiskDrive
State:              Active              Active              Active
Serial Number:      SAK033600AN         SBK044200J8         SAK0403005Q
Prim SW Rev:        2.1(60.8)P1         ---                 ---
Sec SW Rev:         2.1(60.8)P1         ---                 ---
Cur SW Rev:         2.1(60.8)P1         ---                 ---
Boot FW Rev:        2.1(60.8)P1         ---                 ---
800-level Rev:      12                  A0                  06
800-level Part#:    800-05983-01        800-05787-02        800-05052-03
CLEI Code:          0000000000          BA7IBCLAAA          0000000000
Reset Reason:       On Reset From Shell
Card Alarm:         NONE
Failed Reason:      None
Miscellaneous Information:
M8850_LA                          System Rev: 02.01   Sep. 05, 2001 17:18:12 PST
MGX8850                                               Node Alarm: CRITICAL

Crossbar Slot Status:      Present

Alarm Causes
------------
    NO ALARMS
```

Display card details for the current AXSM-16-155.

```
MGX8850.3.AXSM.a > dspcd
                    Front Card          Upper Card          Lower Card
                    ----------          ----------        ------------

Card Type:          AXSM-16-155         MMF-8-155-MT        ---

State:              Active              Present             Absent

Serial Number:      SAK0350008L         SAK0403004A         ---

Boot FW Rev:        2.0(11)A1           ---                 ---

SW Rev:             2.0(1)D             ---                 ---

800-level Rev:      M6                  07                  ---

Orderable Part#:    800-5776-3          800-4819-1          ---

PCA Part#:          73-4504-2           73-3845-1           ---

Reset Reason:On Power up

Card SCT Id: 2

#Lines #Ports #Partitions   #SPVC   #SPVP    #SVC
------ ------ ----------- ------- ------- -------
     2      2           2      10       0      10

Port Group[1]:
#Chans supported:32512  Lines:1.1 1.2 1.3 1.4
Port Group[2]:
#Chans supported:32512  Lines:1.5 1.6 1.7 1.8
Port Group[3]:
#Chans supported:32512  Lines:2.1 2.2 2.3 2.4
Port Group[4]:
#Chans supported:32512  Lines:2.5 2.6 2.7 2.8
```

dspcds

Display Cards

Displays high-level information for all the cards in the node. For more detailed information about a card, execute **dspcd** on the CLI of that card. The information that **dspcds** provides is the:

- Revision level of the boot firmware
- Revision level of the system software
- Date and time of command execution, including GMT offset
- Backplane serial number and its hardware revision level
- The IP address of the statistics master (a workstation)
- Type of card in the front and back slots and the (active/standby) state of each
- Alarm status for each card and the shelf itself
- Redundancy configuration for each slot

Cards on Which This Command Runs

PXM45

Syntax

dspcds

Related Commands

dspcd, **version**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display information for all cards in the MGX 8850 switch.

```
Unknown.7.PXM.a > dspcds
Unknown                          System Rev: 02.00   Aug. 06, 2000 18:03:35 GMT
Backplane Serial No: SAA03270618 Bp HW Rev:  B0      GMT Offset: 0
                                                     Node Alarm: CRITICAL
Card  Front/Back       Card           Alarm      Redundant   Redundancy
Slot  Card State       Type           Status     Slot        Type
---   ----------       --------       --------   -------     -----

01    Active/Active    AXSM_1OC48     NONE       NA          NO REDUNDANCY
02    Active/Active    AXSM_1OC48     NONE       NA          NO REDUNDANCY
03    Active/Active    AXSM_16OC3     NONE       04          PRIMARY SLOT
04    Standby/Active   AXSM_16OC3     NONE       03          SECONDARY SLOT
05    Active/Active    AXSM_4OC12     NONE       NA          NO REDUNDANCY
06    Active/Active    AXSM_4OC12     NONE       NA          NO REDUNDANCY
07    Active/Active    PXM45          NONE       08          PRIMARY SLOT
08    Standby/Active   PXM45          NONE       07          SECONDARY SLOT
09    Active/Active    AXSM_16T3E3    NONE       NA          NO REDUNDANCY
10    Active/Active    AXSM_16T3E3    NONE       NA          NO REDUNDANCY
11    Empty            ---            ---        ---         ---
12    Active/Active    AXSM_2OC12     NONE       NA          NO REDUNDANCY
13    Empty            ---            ---        ---         ---
14    Empty            ---            ---        ---         ---
```

dspcdsct

Display Card SCT

Displays the contents of a card-level service class template (SCT) file. For information about SCTs, see the **cnfcdsct** description. To see the number of the current SCT for the card, use **dspcd**. The examples in this description illustrate the contents of SCT number 2 and SCT 3.

Note Currently, the system does not support certain parameters in the service class templates (SCTs), so you can specify them through **addcon**, **cnfcon**, or Cisco WAN Manager. These parameters are (when applicable) PCR, SCR, and ICR.

Cards on Which This Command Runs

AXSM

Syntax

dspcdsct *<parameter_group>* **[|bw|gen|cosb|vcThr|cosThr]**

Syntax Description

parameter_group can be:

- **bw** for bandwidth
- **gen** for policing and CAC
- **cosb** for Class of Service Buffer specifications
- **vcThr** for VC thresholds
- **cosThr** for COSB thresholds

Related Commands

cnfcdsct, **dspsct**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example SCT 2

This example shows all parameters for SCT 2. Each display consists of one member of the SCT parameter group. The screen examples show the SCT ID that you have displayed (the command itself does not require the SCT ID because it is card-level.

Display the bandwidth parameters for SCT 2.

```
MGX8850.1.AXSM.a > dspcdsct bw
+-----------------------------------------------------------------------------+
Service Class Template [2] : Bw and Policing Parameters
+-----------------------------------------------------------------------------+
| SERV-TYPE |   PCR    |   SCR    |   MCR    |   MBS    |   CDVT   |   ICR    |
+-----------------------------------------------------------------------------+
| VSI-SIG   | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| CBR.1     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| VBR-RT.1  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.2  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.3  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.1 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.2 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.3 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| UBR.1     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| UBR.2     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| ABR       | 00000010 | 00000000 | 01000000 | 00000001 | 00250000 | 00000000 |
| CBR.2     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| CBR.3     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
+-----------------------------------------------------------------------------+
```

Display the policing and CAC parameters (parameter “gen”) for SCT 2. To confirm that the current card-level SCT is SCT 2, execute dspcd.

```
MGX8850.1.AXSM.a > dspcdsct gen
+-------------------------------------------------------------------------------------------------+
Service Class Template [2] : General Parameters
+-------------------------------------------------------------------------------------------------+
| SERV-TYPE | COSB_NUM | CAC_TYPE | UPC_ENB  | CLP-SELEC |    GCRA-1    |    GCRA-2    | CI-CNTRL |
+-------------------------------------------------------------------------------------------------+
| VSI-SIG   | 00000016 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| CBR.1     | 00000003 |    B-CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.1  | 00000004 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.2  | 00000004 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.3  | 00000004 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
| VBR-nRT.1 | 00000005 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.2 | 00000005 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.3 | 00000005 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
| UBR.1     | 00000006 |  LCN_CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | DISABLED |
| UBR.2     | 00000006 |  LCN_CAC |GCRA1-ENB | 000000003 | DSCD/SET-CLP |      DISCARD | DISABLED |
| ABR       | 00000001 |    B-CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | ENABLED  |
| CBR.2     | 00000003 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| CBR.3     | 00000003 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
+-------------------------------------------------------------------------------------------------+
```

Display the Class of Service Buffer parameters for SCT 2. Note the following:

- Min-Rate and Max-Rate do not apply in the current product.
- Excess-Priority is a scheme for distributing excess bandwidth. The lowest number is the highest priority for a connection to receive excess bandwidth. If two or more connections have equal priority, the excess bandwidth is equally distributed between them.
- Explicit Rate Stamping (ERS) applies to only ABR connections.

Cell loss ratio (CLR) is currently hard-coded, so do not attempt to modify it through the Cisco WAN Manager application or the CLI commands.

```
MGX8850.1.AXSM.a > dspcdsct cosb
+-----------------------------------------------------------------------------------+
|Service Class Template [02] : COSB Parameters
+-----------------------------------------------------------------------------------+
| COSB | MIN-RATE | MAX-RATE | MIN-PRIORITY | EXCESS-PRIORITY | ERS ENABLE |  CLR   |
+-----------------------------------------------------------------------------------+
| 0001 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0002 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0003 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-10 |
| 0004 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-08 |
| 0005 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-06 |
| 0006 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0007 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0008 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0009 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0010 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0011 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0012 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0013 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0014 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0015 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0016 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-06 |
+-----------------------------------------------------------------------------------+
```

Display VC thresholds for SCT 2. Note the following:

The Scaling COSB value applies to congestion in a Class of Service Buffer: if a buffer gets congested, this factor determines how quickly the node throttles back the rate at which cells enter the buffer.

The Scaling Log-If is a scaling factor that applies to congestion on a port: when the port gets congested, this factor determines the rate at which the node throttles back traffic (until the port is uncongested).

```
MGX8850.1.AXSM.a > dspcdsct vcThr
+-------------------------------------------------------------------------------------------------------------+
Service Class Template [2] : VC Threshold Parameters
+-------------------------------------------------------------------------------------------------------------+
| SERV-TYPE | VC THRESH | PACKET |  MAX_CELL  |  EFCI   | CLP_HI  |  EPD0   | CLP_LO  | SCALING | SCALING |
|           | TBL IDX   |  MODE  |   THRESH   |         |         |         |  EPD1   |  COSB   |  Log-If |
+-------------------------------------------------------------------------------------------------------------+
| VSI-SIG   |    225    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| CBR.1     |    226    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| VBR-RT.1  |    227    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.2  |    228    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.3  |    229    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.1 |    230    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.2 |    231    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.3 |    232    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| UBR.1     |    233    |   DSB  | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| UBR.2     |    234    |   DSB  | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| ABR       |    235    |   DSB  | 0000050000 | 0200000 | 0800000 | 0600000 | 0800000 | 0000003 | 0000003 |
| CBR.2     |    236    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| CBR.3     |    237    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
```

```
+-------------------------------------------------------------------------------------------------------------+
```

Display the Class of Service Thresholds for SCT 2.

Note The two *random early discard* parameters (RED Factor and RED Prob) have no application in the current release of the product

```
MGX8850.1.AXSM.a > dspcdsct cosThr
+-----------------------------------------------------------------------------------------------+
Service Class Template [00002] : COSB Threshold Parameters
+-----------------------------------------------------------------------------------------------+
| COSB |COSB THRESH|  MAX_CELL   |  EFCI   | CLP_HI  |  EPD0   | CLP_LO  |   RED | RED PROB  |
|      | TBL IDX   |   THRESH    |         |         |         |  EPD1   | |  FACTOR   |
+-----------------------------------------------------------------------------------------------+
| 0001 |  0000114  |  1000000    | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0002 |  0000115  |  1000000    | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0003 |  0000116  |     5000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0004 |  0000117  |    10000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0005 |  0000118  |    50000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0006 |  0000119  |   100000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0007 |  0000120  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0008 |  0000121  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0009 |  0000122  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0010 |  0000123  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0011 |  0000124  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0012 |  0000125  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0013 |  0000126  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0014 |  0000127  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0015 |  0000128  |  1000000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0016 |  0000129  |    10000    | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
+-----------------------------------------------------------------------------------------------+
```

Example SCT 3

This example shows all parameters for SCT 3. Each display consists of one member of the SCT parameter group.

Display the bandwidth parameters for SCT 3.

```
MGX8850.9.AXSM.a > dspcdsct bw
+-------------------------------------------------------------------------------+
Service Class Template [3] : Bw and Policing Parameters
+-------------------------------------------------------------------------------+
| SERV-TYPE |   PCR    |   SCR    |   MCR    |   MBS    |   CDVT   |   ICR    |
+-------------------------------------------------------------------------------+
| VSI-SIG   | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| CBR.1     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| VBR-RT.1  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.2  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.3  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.1 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.2 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.3 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| UBR.1     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| UBR.2     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| ABR       | 00000010 | 00000000 | 01000000 | 00000001 | 00250000 | 00000000 |
| CBR.2     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| CBR.3     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
+-------------------------------------------------------------------------------+
```

Display the general parameters for SCT 3.

```
MGX8850.9.AXSM.a > dspcdsct gen
+----------------------------------------------------------------------------------------------------+
Service Class Template [3] : General Parameters
+----------------------------------------------------------------------------------------------------+
| SERV-TYPE | COSB_NUM | CAC_TYPE | UPC_ENB  | CLP-SELEC |    GCRA-1    |    GCRA-2    | CI-CNTRL |
+----------------------------------------------------------------------------------------------------+
| VSI-SIG   | 00000016 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| CBR.1     | 00000003 |    B-CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.1  | 00000004 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.2  | 00000004 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.3  | 00000004 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
| VBR-nRT.1 | 00000005 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.2 | 00000005 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.3 | 00000005 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
| UBR.1     | 00000006 |  LCN_CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | DISABLED |
| UBR.2     | 00000006 |  LCN_CAC |DISABLED  | 000000003 | DSCD/SET-CLP |      DISCARD | DISABLED |
| ABR       | 00000001 |    B-CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | ENABLED  |
| CBR.2     | 00000003 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| CBR.3     | 00000003 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
+----------------------------------------------------------------------------------------------------+
```

Display the Class of Service Buffer parameters for SCT 3 Note the following:

- Min-Rate and Max-Rate do not apply in the current product.
- Excess-Priority is a scheme for distributing excess bandwidth. The lowest number is the highest priority for a connection to receive excess bandwidth. If two or more connections have equal priority, the excess bandwidth is equally distributed between them.
- Explicit Rate Stamping (ERS) applies to only ABR connections.
- Cell loss ratio (CLR) is currently hard-coded, so do not attempt tot modify it through the Cisco WAN Manager application or the CLI commands.

```
MGX8850.9.AXSM.a > dspcdsct cosb
+--------------------------------------------------------------------------------------+
|Service Class Template [03] : COSB Parameters
+--------------------------------------------------------------------------------------+
| COSB | MIN-RATE | MAX-RATE | MIN-PRIORITY | EXCESS-PRIORITY | ERS ENABLE |  CLR   |
+--------------------------------------------------------------------------------------+
| 0001 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0002 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0003 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-10 |
| 0004 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-08 |
| 0005 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-06 |
| 0006 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0007 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0008 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0009 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0010 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0011 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0012 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0013 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0014 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0015 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0016 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-06 |
+-------------------------------------------------------------------------------------
```

dspcon

Display Connection

Display information about an SPVC. The contents of the display on the AXSM and the PXM45 differ slightly. On both cards, the **dspcon** output appears in sections to make the information easier to sort.

Most of the information in the **dspcon** output comes from **addcon** execution. See the **addcon** description for more information. Also, executing **cnfpnni-intf** can affect the **dspcon** output.

Cards on Which This Command Runs

PXM45, AXSM

Display Connection on the PXM45

On the PXM45, **dspcon** shows the following connection identifiers:

- NSAP address, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type and compliance (for example, UBR for service type and UBR.1 for ATM Forum compliance).
- Bearer class (relates to voice traffic and is reserved for future use).
- Whether continuity checking or frame discard are enabled (see **addcon** description).
- Cause of the last failure. This field can also show that no errors have occurred since the connection was first added by displaying "SPVC Established." If a failure occurred, the Attempts field shows the number of times the system attempted to re-establish service. If no failures have occurred, the Attempts field contains a 0.
- L-Util and R-Util are the local and remote percent of utilization assigned to the connection. Currently, the default of 100% is the only value.
- Cost values for the connection's route: the two fields in this category are Max Cost and Routing Cost. The Max Cost is a cost-per-link configured for a service type (such as UBR) through the **cnfpnni-intf** command. When you add the SPVC through **addcon**, you can specify a maximum routing cost through the maximum cost (maxcost) parameter. The maxcost represents the maximum cost for an individual connection. The system uses the cost-per-link for the service type and the maxcost for the connection to determine whether a route costs too much. After the system creates a route, the total number of links yields the Routing Cost.

  The default cost-per-link is 5040, so if a particular service type uses the default and a route consists of 4 links, the Routing Cost is 20160. If the **dspspvc** display shows that Max Cost is –1, no limit was specified through **cnfpnni-intf**, and the resulting Routing Cost is 0.
- Broadcast type: point-to-point or multicast.

The Traffic Parameters section shows the standard parameters PCR, SCR, and CDV in the receive and transmit directions.

Display Connection on the AXSM

On the AXSM, **dspcon** shows the following connection identifiers:

- NSAP address, logical port, VPI/VCI, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type (for example, ABR).
- A number indicating the controller. For example, 2 refers to PNNI. The **addcontroller** command specifies the controller.
- The administrative state is either up or down. This state results from **addcon** or **dncon**/**upcon**. Note that, after you down a connection with at the connection master endpoint, the **dspcon** command shows the connection as “down” when you execute it at the master endpoint and “failed” when you execute it at the slave endpoint. (See also **dncon** description).
- The operational state is either OK or failed. The operational state can apply to a connection regardless of the administrative state.

The traffic management parameters consist of:

- Local and remote UPC parameters of PCR, MBS, CTD, CDVT, and so on. A –1 in a field means that the parameter was not specified. The characters “N/A” indicate that the parameter does not apply to the service type.

These other fields also pertain to connection integrity:

- OAM connectivity check enable or disable.
- Loopback test enable/disable and loopback type.
- Round trip delay in microseconds. This field is non-zero only if you previously executed **tstdelay**.

The **dspcon** command requires a unique connection identifier. If you do not have the information to identify a connection, execute **dspcons**. On the AXSM, **dspcons** identifies all the connections on the AXSM. On the PXM45, **dspcons** identifies all the connections on the node. (See **dspcons** description).

Syntax

On the AXSM

`dspcon` *`<ifNum> <vpi> <vci>`*

On the PXM45:

`dspcon` *`<portid> <vpi> <vci>`*

Syntax Description

ifNum	(AXSM) Logical interface (port) number. For AXSM, the range is 1–60.
portid	(PXM45) The portid has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*], but *shelf* currently is always 1 and optional.
vpi	VPI number. At the UNI, the range is 0–255. At the NNI, the range is 0–4095.
vci	VCI number. For a VCC, the range is 32–65535. For a VPC, the VCI is 0.

Related Commands

addcon, **dspcons**, **cnfcon**

PXM45 Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active	Privilege: GROUP1

Examples

Display connection 5 31 63000 on the current AXSM.

```
MGX8850.1.AXSM.a > dspcon 5 31 63000
--------------------------------------------------------------------------

Local   :               NSAP  Address               port     vpi    vci

(M)       47009181000000000107BE92F3F00000101180500  1.01.05 31 63000
Remote  :               NSAP  Address               port     vpi    vci

(S)       47009181000000000107BE92F3F00000101180500  1.01.05   3201 100
--------------------------------------------------------------------------

Conn. Type    :     VCC                          Admn Status  : ADMN-UP
Service Type  :    cbr1                          Oper Status  : FAIL
Controller    :       2
--------------------------------------------------------------------------

Local PCR     :   10000                          Remote PCR   : 1000
Local SCR     :     N/A                          Remote SCR   : N/A
Local CDV     :      -1                          Remote CDV   : -1
Local CTD     :      -1                          Remote CTD   : -1
Local MBS     :     N/A                          Remote MBS   : N/A
Local CDVT    :      -1                          Remote CDVT  : -1
Admin weight  :      -1                          Frame discard: N
--------------------------------------------------------------------------

OAM CC Config :DISABLED                          Statistics   : DISABLED
--------------------------------------------------------------------------

Loopback Type :No  Lpbk | Dir:N/A     | Status: No Lpbk | RTD: 0 us
```

On the CLI of the PXM45, display connection 20 100 on 11:1.1:2.

```
Unknown.7.PXM.a > dspcon 11:1.1:2 20 100
Port                   Vpi Vci                          Owner      State
-------------------------------------------------------------------------
Local  11:1.1:2        20.100                           MASTER     FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00
Remote 11:1.1:2        10.100                           SLAVE      FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: CBR         Conformance: CBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  50             Rx PCR:  50
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

Display information for vpi/vci 10 100 on port ID 1:1.1:1. In this case, port ID and remote and local NSAP addresses are the same, so the connection is a DAXCON. Also, the Max Cost is –1. The Max Cost of -1 means no cost-per-link was specified for UBR service, and therefore the Routing Cost is 0.

```
node19.8.PXM.a > dspcon 1:1.1:1 10 100
Port                   Vpi Vci                          Owner      State
-------------------------------------------------------------------------
Local  1:1.1:1         10.100                           SLAVE      OK
       Address: 47.00918100000000001a53c82d.000001011801.00
Remote 1:1.1:1         11.101                           MASTER     OK
       Address: 47.00918100000000001a53c82d.000001011801.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: UBR         Conformance: UBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  14             Rx PCR:  14
Tx SCR:  3              Rx SCR:  3
Tx MBS:  1              Rx MBS:  1
Tx CDVT: -1             Rx CDVT: -1
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

dspcons

Display Connections—display basic information for all connections.

The default usage of **dspcons** uses no parameters and causes all available information for the connections to appear. To narrow the scope of the output, use one or more optional parameters.

The **dspcons** command runs on the CLI of either the AXSM or the PXM45. The set of optional parameters and the output are different on these cards. (See Syntax Description for the card-specific parameters.) On the AXSM, the columns at the head of the information fields are:

record	A number for the connection with internal application only. It resides in the database on the AXSM and is not affected by user input. The system creates this number when you create the connection. The Cisco WAN Manager application uses this number.
Identifier	Identifies the connection in the format *port vpi vci*.
Type	Shows whether the connection is a VCC or a VPC.
SrvcType	The service type—VBR, and so on. (See **addcon** description).
M/S	Indicates whether the endpoint specified by *Identifier* is the master or slave.
Upld	The hexadecimal Upload number is an encoded timestamp the Cisco WAN Manager application uses to determine when a connection was created or modified. In the CLI context, this field has little meaning.
Adm	The administrative state of the connection. If the connection is down, it may have resulted from the **dncon** command.
Alarm	Shows the alarm status of the connection.

When you execute **dspcons** on the CLI of the PXM45, the output shows:

Local and Remote Port ID	The display contains a column for the local port ID and a column for the remote port ID. The port ID has the format that the network controller utilizes: [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]
Local and Remote VPI.VCI	The VPI and VCI at the local and remote ends of the connection.
State	The State column shows whether the connection is OK, down (by the **dncon** command), failed, or has an alarm indication signal (AIS) or *abit* errors.
Owner	Whether the endpoint is master or slave.
Local and Remote NSAP	An NSAP address for each end of the connection.

Cards on Which This Command Runs

PXM45, AXSM

PXM45 Syntax

dspcons[-**port** *portid*] [-**vpi** *starting-vpi*] [-**vci** *starting vci*]
[-**state** {**fail**|**ais**|**abit**|**ok**|**down**}] [-**owner** {**master**|**slave**}]

PXM45 Syntax Description

-**port**	The port identifier (*portid*) in the format that the network controller utilizes: [*shelf*.]*slot*[:*subslot*].*port*[:*subport*] Currently, the value for *shelf* is always 0 and therefore is not necessary.
-**vpi**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**vci**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**state**	A specific connection state. The display shows only the connections with the state you specify. Note that on the PXM45, you must spell out the entire state keyword. The keywords for specifying a state are **failed**—only failed connections **ais—**connections with alarm indication signal (AIS) set **abit**—connections on which an A-bit error has occurred **ok—**connections with no problems **down**—connections that are administratively down because a user has executed **dncon** to down the connection
-**owner**	Specifies connections where the local endpoint is either master or slave.

Related Commands

dspcon, **addcon**, **cnfcon**, **delcon**, **dncon**, **upcon**

PXM45 Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active, standby	Privilege: GROUP1

PXM45 Example

Display all connections by entering **dspcons** on the CLI of the PXM45.

```
MGX8850.7.PXM.a > dspcons

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
3:1.1:1            20 0      6:1.1:1            20 0       OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001031801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
5:1.1:1           100 100    5:1.1:1           100 200     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
5:1.1:1           100 200    5:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
6:1.1:1            20 0      3:1.1:1            20 0       OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001031801.00
6:1.1:1           100 100    6:1.1:1           100 200     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           100 200    6:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK          SLAVE

6:1.1:1           100 200    6:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK          SLAVE

Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061803.00
6:2.1:3           200 200    6:1.1:1           200 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061803.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
9:1.3:3            10 100    Routed              0 0       FAIL        SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001091803.00
Remote Addr: 00.000000000000000000000000.000000000000.00
11:1.1:2           10 100    11:1.1:2           20 100     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
11:1.1:2           20 100    11:1.1:2           10 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00
```

AXSM Example

Display all connections on the current AXSM. In this example, only one connection exists. Master and slave endpoints are shown.

```
GN.6.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld    Admn   Alarm
------    ----------   ----   --------   ---    ----    ----   -----
    0  01.0010.00100   VCC        cbr1   S   010c7953    UP      none
    1  04.0020.00100   VCC        cbr1   M   010c7964    UP      none
```

dspcontrollers

Display Controllers

Displays all controllers that have been added through the **addcontroller** command.

Cards on Which This Command Runs

PXM45

Syntax

`dspcontrollers`

Syntax Description

This command takes no parameters.

Related Commands

addcontroller, **delcontroller**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display all controller. In this example, the switch has only one controller—PNNI. The display also shows that the controller is internal (slot 7) and has the optional, user-specified name "PNNITWO." Apart from controller information, the display shows that no shelf alarms exist.

```
MGX8850.7.PXM.a > dspcontrollers
MGX8850                          System Rev: 02.00   Jul. 30, 2000 09:39:36 GMT
MGX8850                                               Shelf Alarm: NONE
Number of Controllers:        1
Controller Name:              PNNITWO
Controller Id:                2
Controller Location:          Internal
Controller Type:              PNNI
Controller Logical Slot:      7
Controller Bay Number:        0
Controller Line Number:       0
Controller VPI:               0
Controller VCI:               0
Controller In Alarm:          NO
Controller Error:
```

dspfdr

Display Feeder

Displays the configuration information for the feeder on the specified port (*ifNum*). The interface numbers of active ports are displayed in the **dspports** command report.

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

dspfdr <*ifNum*>

Syntax Description

ifNum	The interface number of the port on which to display the feeder information. The interface numbers of active ports are displayed in the **dspports** command report.

Card(s) on Which This Command Executes

AXSM

Related Commands

addfdr, delfdr, dspfdrs

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

```
MGX8850.1.AXSM.a > dspfdr 4
  Feeder Interface Number        : 4
  Feeder Name                    : MGX8850
  Feeder LAN IP Address          : 192.0.0.0
  Feeder Network IP Address      : 0.0.0.0
  Feeder Remote Shelf            : 1
  Feeder Remote Slot             : 1
  Feeder Remote Port             : 3
  Feeder Type                    : AXSM
  Feeder Model Number            : 8850
  Feeder LMI Configuration       : Up
  Feeder Lmi Link Status         : Up
  Feeder Alarms                  : Clear
```

dspfdrs

Display Feeders

Displays all feeders on all ports on the AXSM card and their information.

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

dspfdrs

Card(s) on Which This Command Executes

AXSM

Related Commands

addfdr, delfdr, dspfdr

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

```
MGX8850.1.AXSM.a > dspfdrs

If  Feeder     Lan              Net              Rmt   Rmt   LMI    LMI   Feeder
    Name       IP               IP               Slot  Port  Admin  Oper  Alarm
--  ---------  ---------------  ---------------  ----  ----  -----  ----  ------
 4   MGX8850       192.0.0.0          0.0.0.0    1     3    Up     Up    Clear
```

dspfdrstat

Display Feeder Statistics

Displays the LMI and node statistics for the feeder on the specified port (*ifNum*). The interface numbers of active ports are displayed in the **dspports** command report.

For more detailed information on configuring a feeder, see the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

Syntax

```
dspfdrstat <ifNum>
```

Syntax Description

ifNum	The interface number of the port on which to display the feeder statistics. The interface numbers of active ports are displayed in the **dspports** command report.

Card(s) on Which This Command Executes

AXSM

Related Commands

clrfdrstat

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

```
MGX8850.1.AXSM.a > dspfdrstat 7
```

dspln

Display Line

Display the characteristics of a physical line.

Note The connection count includes control VCs when you use **dspln** on a service module. However, when you use **dspcd** or **dsppnport**(**s**) on the CLI of the PXM, the display does not include control VCs.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dspln <-ds3 | -sonet> *<bay.line>*

AXSM Syntax Description

-ds3	Command delineator that precedes the *bay.line* number entry for a T3 line.
-sonet	Command delineator that precedes the *bay.line* number entry for a SONET line.
bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

upln, **cnfln**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display T3 line 1 on the current AXSM.

```
MGX8850.11.AXSM.a > dspln -ds3 1.1
  Line Number         : 1.1
  Admin Status        : Down              Alarm Status        : Clear
  Line Type           : ds3cbitadm        Number of ports     : 0
  Line Coding         : ds3B3ZS           Number of partitions: 0
  Line Length(meters) : 0                 Number of SPVC      : 0
  OOFCriteria         : 3Of8Bits          Number of SPVP      : 0
  AIS c-Bits Check    : Check             Number of SVC       : 0
  Loopback            : NoLoop
  Xmt. Clock source   : localTiming
  Rcv FEAC Validation : 4 out of 5 FEAC codes
```

Display OC-48 line on the current OC-12 AXSM.

```
MGX8850.1.AXSM.a > dspln -sonet 2.1
  Line Number           : 2.1
  Admin Status          : Up                  Alarm Status        : Clear
  Loopback              : NoLoop              APS enabled         : Disable
  Frame Scrambling      : Enable              Number of ports     : 1
  Xmt Clock source      : localTiming         Number of partitions: 1
  Line Type             : sonetSts12c         Number of SPVC      : 0
  Medium Type(SONET/SDH) : SONET              Number of SVC       : 4
  Medium Time Elapsed    : 506223
  Medium Valid Intervals : 96
  Medium Line Type       : ShortSMF
```

Note When APS is enabled, the alarm status line shows the alarm status of the active line.

dsplns

Display Lines

Displays the configuration for all lines on a card. For each line, the output information consists of the:

- Bay and line number
- Line state—up (active) or down (inactive)
- The line type
- Whether any loopback currently exists on the line
- Line coding
- Frame scrambling status (enabled or disabled)
- Configured line length in meters (applies to only T3 or E3)
- Criteria for Out of Frame (OOF) error (applies to only T3 or E3)
- Whether C-bit (AIS) checking is enabled (applies to only T3 or E3)
- The medium line type—long reach, single-mode fiber, for example
- The alarm status—clear, critical, and so on

For information on an individual line, use **dspln**. Also, the **dspln** command shows the transmit clock configuration if one exists.

Note Only lines that have *Admin* status up are displayed.

Cards on Which This Command Runs

AXSM

Syntax

`dsplns`

Related Commands

cnfln, **dspcds**, **dspln**, **upln**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the configuration of the lines on an AXSM-4-622.

```
MGX8850.1.AXSM.a > dsplns
                                            Medium Medium
  Sonet  Line     Line     Line    Frame    Line   Line     Alarm   APS
  Line  State     Type     Lpbk   Scramble Coding Type     State   Enabled
  ----- ----- ------------ ------ -------- ------ -------  -----   --------
   1.1     Up sonetSts12c NoLoop   Enable  Other ShortSMF Critical Disable
   1.2     Up sonetSts12c NoLoop   Enable  Other ShortSMF    Clear Disable
   2.1     Up sonetSts12c NoLoop   Enable  Other ShortSMF    Clear Disable
   2.2     Up sonetSts12c NoLoop   Enable  Other ShortSMF    Clear Disable
```

Display line configuration on the current AXSM-1-2488.

```
MGX8850.1.AXSM.a > dsplns
                                              Medium Medium
  Sonet  Line     Line     Line    Frame      Line   Line
  Line  Status    Type     Lpbk   Scramble   Coding Type
  ----- ----- ------------ ------  -------- -------- -------
   1.1   Down sonetSts48c NoLoop   Enable  Other ShortSingleMode
```

Display the configuration of each T3 line on the current AXSM-16-T3E3.

```
MGX8850.7.AXSM.a > dsplns
  Line  Line      Line        Line     Length  OOF      AIS
  Num   State     Type        Lpbk    (meters) Criteria c-BitsCheck
  ------------ ----------- ----------- ------ -------- ---------
   1.1    Up  ds3m23plcp PayloadLoop       0  fBits3Of8      Check
   1.2  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.3  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.4  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.5  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.6  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.7  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   1.8  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.1  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.2  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.3  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.4  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.5  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.6  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.7  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
   2.8  Down  ds3cbitadm      NoLoop       0  fBits3Of8      Check
```

dspload

Display Load

Display the current level of usage of various parameters on a partition. To convey a picture of what is available on a resource partition, the display shows the configured bandwidth and connection numbers and what has actually been utilized.

Cards on Which This Command Runs

AXSM

Syntax

`dspload` `<`*`ifNum`*`> <`*`partId`*`>`

Syntax Description

ifNum	Logical interface (port) number. For AXSM, the range is 1–60.
partId	The partition identifier. The range is 1–5. If necessary, use **dsprscprtns** to see the existing partitions.

Related Commands

dsprscprtn, **addcon**, **dspcons**, **dspcon**, **cnfcon**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the load on partition number 1 on logical port 1. The display shows that very little of the available connections and bandwidth have been used. Also, no exceptions have been recorded.

```
M8850_NY.1.AXSM.a > dspload 1 1
        +---------------------------------------------+
        |   I N T E R F A C E   L O A D   I N F O   |
        +---------------------------------------------+
        | Maximum Channels         : 0001000         |
        | Guaranteed Channels      : 0000100         |
        | Igr Maximum Bandwidth    : 1412830         |
        | Igr Guaranteed Bandwidth : 1412830         |
        | Egr Maximum Bandwidth    : 1412830         |
        | Egr Guaranteed Bandwidth : 1412830         |
        | Available Igr Channels   : 0000999         |
        | Available Egr Channels   : 0000999         |
        | Available Igr Bandwidth  : 1410830         |
        | Available Egr Bandwidth  : 1410830         |
        +---------------------------------------------+
        |          E X C E P T -- V A L U E S        |
        +---------------------------------------------+
        | SERV-CATEG | VAR-TYPE | INGRESS  | EGRESS  |
        | VSI-SIG    | Avl Chnl | 0000999  | 0000999 |
        | CBR        | Avl Chnl | 0000999  | 0000999 |
        | VBR-RT     | Avl Chnl | 0000999  | 0000999 |
        | VBR-nRT    | Avl Chnl | 0000999  | 0000999 |
        | UBR        | Avl Chnl | 0000999  | 0000999 |
        | ABR        | Avl Chnl | 0000999  | 0000999 |
        +---------------------------------------------+
        | VSI-SIG    | Avl Bw   | 1410830  | 1410830 |
        | CBR        | Avl Bw   | 1410830  | 1410830 |
        | VBR-RT     | Avl Bw   | 1410830  | 1410830 |
        | VBR-nRT    | Avl Bw   | 1410830  | 1410830 |
        | UBR        | Avl Bw   | 1410830  | 1410830 |
        | ABR        | Avl Bw   | 1410830  | 1410830 |
        +---------------------------------------------+
```

dsppart

Display Resource Partition

Displays information about one resource partition. The displayed information is shown in the example.

Note The **dsppart** and **dsprscprtn** commands are identical. The name 'dsprscprtn' is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Note The connection count includes control VCs when you execute **dsppart** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

The total number of connections in the display includes control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that do not include control VCs, use **dsppnport**.

Cards on Which This Command Runs

AXSM

Syntax

`dsppart <ifNum> <partId>`

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_id	Partition identifier in the range 1–5.

Related Commands

addpart, **cnfpart**, **delpart**, **dspparts**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display resource partition 1 on logical port 1 of the current AXSM.

```
MGX8850.1.AXSM.a > dsppart 1 1
  Interface Number              : 1
  Partition Id                  : 1        Number of SPVC: 0
  Controller Id                 : 2        Number of SPVP: 0
  egr Guaranteed bw(.0001percent): 1000000  Number of SVC : 2
  egr Maximum bw(.0001percent)   : 1000000
  ing Guaranteed bw(.0001percent): 1000000
  ing Maximum bw(.0001percent)   : 1000000
  min vpi                       : 0
  max vpi                       : 4095
  min vci                       : 33
  max vci                       : 65535
  guaranteed connections        : 1000
  maximum connections           : 32000
```

dspparts

Display Resource Partitions

Display information for all the resource partitions on the current card. The displayed information is shown in the example.

For information on specific elements of a resource partition, see the description of **addpart**.

Note The **dspparts** and **dsprscprtns** commands are identical. The name 'dsprscprtns' is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

`dspparts`

Related Commands

addpart, **delpart**, **cnfpart**, **dsppart**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display all resource partitions on the current AXSM card.

```
MGX8850.1.AXSM.a > dspparts
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
----------------------------------------------------------------------------
 1   1    2 1000000 1000000 1000000 1000000    0 4095    33 65535   1000  32000
 2   1    2 1000000 1000000 1000000 1000000    0  255    33 65535   1000  32000
20   1    2 1000000 1000000 1000000 1000000    1    1    33 65535      2    512
21   1    2 1000000 1000000 1000000 1000000    0  255    33 65535      2    512
22   1    2 1000000 1000000 1000000 1000000    0  255    33 65535      2    512
23   1    2 1000000 1000000 1000000 1000000  255  255    33 65535      2    512
```

dspport

Display Port

Displays the configuration for a logical port. The displayed information is shown in the example. For a description of each item, see **addport**.

The total number of connections in the display includes control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that do not include control VCs, use **dsppnport**.

Note The SCT ID that **dspport** shows pertains to the port. For the card-level SCT ID, use **dspcd**.

Cards on Which This Command Runs

AXSM

Syntax

```
dspport <ifNum>
```

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60.

Related Commands

addport, **dnport**, **dspports**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the port configuration for logical port 2 on the current AXSM.

```
MGX8850.1.AXSM.a > dspport 2
  Interface Number               : 2
  Line Number                    : 2.1
  Admin State                    : Up       Operational State   : Up
  Guaranteed bandwidth(cells/sec): 100000    Number of partitions: 1
  Maximum bandwidth(cells/sec)   : 100000    Number of SPVC      : 0
  ifType                         : NNI       Number of SVC       : 4
  SCT Id                         : 3
  VPI number(VNNI only)          : 0
```

dspports

Display Ports

Displays general information about all logical ports on the card. On the AXSM, the information consists of the following for each logical port:

- Logical port number (*ifNum*). On the AXSM, for example, the range is 1–60.
- Physical line number in the format *bay.port*.
- Operation status—whether the port is up or down.
- The minimum guaranteed rate in cells per second.
- The maximum allowed rate for the port in cells per second.
- The ID of the port-level SCT (see **addport**).
- The VPI number (applies only where virtual NNIs are available).

Cards on Which This Command Runs

AXSM

Syntax

```
dspports
```

Related Commands

addport, **cnfport**, **delport**, **dspport**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display the logical ports on the current AXSM. Note that if no SCT file had been specified for a particular logical port (*ifNum*), the output gives a warning about the default SCT file usage for that port. See also the description for the **cnfcdsct** command.

```
MGX8850.1.AXSM.a > dspports

ifNum Line Oper.  Guaranteed Maximum       sctID               ifType   VPI
           Status Rate       Rate                                   (VNNI only)
----- ---- ------ ---------- --------- ---------------------   ------ ---------
    1   1.1     Up      10000     10000   4                        UNI    0
    2   1.2   Down      10000     10000   4                        UNI    0
    3   1.3   Down      10000     10000   4                       VNNI    1
    4   1.3   Down      10000     10000   0 !DefaultSCT used!     VNNI    2 UNI    0
```

dspportsct

Display Port SCT

Displays the contents of the service class template (SCT) on a port.

Note Currently, the system does not support certain parameters in the service class templates (SCTs), so you can specify them through **addcon**, **cnfcon**, or Cisco WAN Manager.
These parameters are (when applicable) PCR, SCR, and ICR.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dspportsct *<parameter_group> <ifnum>*

AXSM Syntax Description

parameter_group	An aspect of the SCT for display: • **bw** for bandwidth • **gen** for policing and CAC • **cosb** • **vcThr** for VC thresholds • **cosThr** for COSB thresholds
ifnum	Logical interface (port) number. For AXSM, the range is 1–60.

Related Commands

addport, **cnfport**, **delport**, **dspport**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples SCT 2

This section displays all parameters for port SCT 2.

```
MGX8850.1.AXSM.a > dspportsct bw 2
+-----------------------------------------------------------------------------+
Service Class Template [2] : Bw and Policing Parameters
+-----------------------------------------------------------------------------+
| SERV-TYPE |    PCR    |    SCR    |    MCR    |    MBS    |   CDVT    |    ICR    |
+-----------------------------------------------------------------------------+
| VSI-SIG   | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| CBR.1     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| VBR-RT.1  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.2  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.3  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.1 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.2 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.3 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| UBR.1     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| UBR.2     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| ABR       | 00000010 | 00000000 | 01000000 | 00000001 | 00250000 | 00000000 |
| CBR.2     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| CBR.3     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
+-----------------------------------------------------------------------------+
```

Display the policing and CAC parameters (parameter “gen”) for SCT 2. To confirm that the current card-level SCT is SCT 2, run the **dspcd** command.

```
MGX8850.1.AXSM.a > dspportsct gen 2
+-----------------------------------------------------------------------------------------------+
Service Class Template [2] : General Parameters
+-----------------------------------------------------------------------------------------------+
| SERV-TYPE | COSB_NUM | CAC_TYPE | UPC_ENB   | CLP-SELEC |    GCRA-1    |    GCRA-2    | CI-CNTRL |
+-----------------------------------------------------------------------------------------------+
| VSI-SIG   | 00000016 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| CBR.1     | 00000003 |    B-CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.1  | 00000004 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.2  | 00000004 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.3  | 00000004 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
| VBR-nRT.1 | 00000005 |    B-CAC |GCRA 1 & 2| 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.2 | 00000005 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.3 | 00000005 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
| UBR.1     | 00000006 |  LCN_CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | DISABLED |
| UBR.2     | 00000006 |  LCN_CAC |GCRA1-ENB | 000000003 | DSCD/SET-CLP |      DISCARD | DISABLED |
| ABR       | 00000001 |    B-CAC |GCRA1-ENB | 000000003 |      DISCARD |      DISCARD | ENABLED  |
| CBR.2     | 00000003 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      DISCARD | DISABLED |
| CBR.3     | 00000003 |    B-CAC |GCRA 1 & 2| 000000001 |      DISCARD |      SET-CLP | DISABLED |
+-----------------------------------------------------------------------------------------------+
```

Display the Class of Service Buffer parameters for SCT 2 Note the following:

- Min-Rate and Max-Rate do not apply in the current product.
- Excess-Priority is a scheme for distributing excess bandwidth. The lowest number is the highest priority for a connection to receive excess bandwidth. If two or more connections have equal priority, the excess bandwidth is equally distributed between them.
- Explicit Rate Stamping (ERS) applies to only ABR connections.

Cell loss ratio (CLR) is currently hard-coded, so do not attempt tot modify it through the Cisco WAN Manager application or the CLI commands.

```
MGX8850.1.AXSM.a > dspportsct cosb 2
+-----------------------------------------------------------------------------------+
|Service Class Template [02] : COSB Parameters

+-----------------------------------------------------------------------------------+
| COSB | MIN-RATE | MAX-RATE | MIN-PRIORITY | EXCESS-PRIORITY | ERS ENABLE |  CLR   |
+-----------------------------------------------------------------------------------+
| 0001 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0002 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0003 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-10 |
| 0004 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-08 |
| 0005 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-06 |
| 0006 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0007 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0008 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0009 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0010 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0011 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0012 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0013 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0014 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0015 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0016 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-06 |
+-----------------------------------------------------------------------------------+
```

Display VC thresholds for SCT 2. Note the following:

The Scaling COSB value applies to congestion in a Class of Service Buffer: if a particular buffer becomes congested, this scaling factor determines the how quickly the rate at which cells enter the buffer is throttled back (until the buffer is no longer congested, at which time normal rates resume).

The Scaling Log-If is a scaling factor that applies to congestion on an entire port: when the whole port is congested, this factor determines the rate at which traffic is throttled back (until the port is no longer congested, at which time normal rates resume).

```
MGX8850.1.AXSM.a > dspportsct vcThr 2
+-----------------------------------------------------------------------------------------------------------+
Service Class Template [2] : VC Threshold Parameters
+-----------------------------------------------------------------------------------------------------------+
| SERV-TYPE | VC THRESH | PACKET |  MAX_CELL  |  EFCI   | CLP_HI  |  EPD0   | CLP_LO  | SCALING | SCALING |
|           | TBL IDX   |  MODE  |   THRESH   |         |         |         |  EPD1   |  COSB   |  Log-If |
+-----------------------------------------------------------------------------------------------------------+
| VSI-SIG   |    002    |  DSB   | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| CBR.1     |    003    |  DSB   | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| VBR-RT.1  |    004    |  DSB   | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.2  |    005    |  DSB   | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.3  |    006    |  DSB   | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.1 |    007    |  DSB   | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.2 |    008    |  DSB   | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.3 |    009    |  DSB   | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| UBR.1     |    010    |  DSB   | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| UBR.2     |    011    |  DSB   | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| ABR       |    012    |  DSB   | 0000050000 | 0200000 | 0800000 | 0600000 | 0800000 | 0000003 | 0000003 |
| CBR.2     |    013    |  DSB   | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| CBR.3     |    014    |  DSB   | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001
+-----------------------------------------------------------------------------------------------------------+
```

Display the Class of Service Thresholds for SCT 2.

Note The two *random early discard* parameters (RED Factor and RED Prob) have no application in the current release of the product

```
MGX8850.1.AXSM.a > dspportsct cosThr 2
+-----------------------------------------------------------------------------------------------+
Service Class Template [00002] : COSB Threshold Parameters
+-----------------------------------------------------------------------------------------------+
| COSB |COSB THRESH|  MAX_CELL  |  EFCI   | CLP_HI  |  EPD0   | CLP_LO  |    RED | RED PROB  |
|      | TBL IDX   |   THRESH   |         |         |         |  EPD1   | |  FACTOR   |
+-----------------------------------------------------------------------------------------------+
| 0001 |  0000002  |  1000000   | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0002 |  0000003  |  1000000   | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0003 |  0000004  |     5000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0004 |  0000005  |    10000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0005 |  0000006  |    50000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0006 |  0000007  |   100000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0007 |  0000008  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0008 |  0000009  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0009 |  0000010  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0010 |  0000011  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0011 |  0000012  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0012 |  0000013  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0013 |  0000014  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0014 |  0000015  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0015 |  0000016  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0016 |  0000017  |    10000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
+-----------------------------------------------------------------------------------------------+
```

Example SCT 3

This example shows all parameters for SCT 3. Each display consists of one member of the SCT parameter group.

Display the bandwidth parameters for SCT 3.

```
MGX8850.9.AXSM.a > dspportsct bw 3
+-----------------------------------------------------------------------------+
Service Class Template [3] : Bw and Policing Parameters
+-----------------------------------------------------------------------------+
| SERV-TYPE |   PCR    |   SCR    |   MCR    |   MBS    |   CDVT   |   ICR    |
+-----------------------------------------------------------------------------+
| VSI-SIG   | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| CBR.1     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| VBR-RT.1  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.2  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-RT.3  | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.1 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.2 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| VBR-nRT.3 | 00001000 | 01000000 | 00000000 | 00000050 | 00250000 | 00000000 |
| UBR.1     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| UBR.2     | 00000010 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| ABR       | 00000010 | 00000000 | 01000000 | 00000001 | 00250000 | 00000000 |
| CBR.2     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
| CBR.3     | 00001000 | 00000000 | 00000000 | 00000001 | 00250000 | 00000000 |
+-----------------------------------------------------------------------------+
```

Display the general parameters for SCT 3.

```
MGX8850.9.AXSM.a > dspportsct gen 3
+-------------------------------------------------------------------------------------------------+
Service Class Template [3] : General Parameters
+-------------------------------------------------------------------------------------------------+
| SERV-TYPE | COSB_NUM | CAC_TYPE | UPC_ENB  | CLP-SELEC |    GCRA-1    |    GCRA-2    | CI-CNTRL |
+-------------------------------------------------------------------------------------------------+
| VSI-SIG   | 00000016 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| CBR.1     | 00000003 |    B-CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.1  | 00000004 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.2  | 00000004 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-RT.3  | 00000004 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
| VBR-nRT.1 | 00000005 |    B-CAC |DISABLED  | 000000002 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.2 | 00000005 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| VBR-nRT.3 | 00000005 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
| UBR.1     | 00000006 |  LCN_CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | DISABLED |
| UBR.2     | 00000006 |  LCN_CAC |DISABLED  | 000000003 | DSCD/SET-CLP |      DISCARD | DISABLED |
| ABR       | 00000001 |    B-CAC |DISABLED  | 000000003 |      DISCARD |      DISCARD | ENABLED  |
| CBR.2     | 00000003 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      DISCARD | DISABLED |
| CBR.3     | 00000003 |    B-CAC |DISABLED  | 000000001 |      DISCARD |      SET-CLP | DISABLED |
+-------------------------------------------------------------------------------------------------+
```

Display the Class of Service Buffer parameters for SCT 3 Note the following:

- Min-Rate and Max-Rate do not apply in the current product.
- Excess-Priority is a scheme for distributing excess bandwidth. The lowest number is the highest priority for a connection to receive excess bandwidth. If two or more connections have equal priority, the excess bandwidth is equally distributed between them.
- Explicit Rate Stamping (ERS) applies to only ABR connections.
- Cell loss ratio (CLR) is currently hard-coded, so do not attempt to modify it through CWM.

```
MGX8850.9.AXSM.a > dspportsct cosb 3
+-----------------------------------------------------------------------------------+
|Service Class Template [03] : COSB Parameters
+-----------------------------------------------------------------------------------+
| COSB | MIN-RATE | MAX-RATE | MIN-PRIORITY | EXCESS-PRIORITY | ERS ENABLE |  CLR   |
+-----------------------------------------------------------------------------------+
| 0001 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0002 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0003 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-10 |
| 0004 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-08 |
| 0005 | 00000000 | 00000100 |          000 |             001 |    DISABLE | 10^-06 |
| 0006 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0007 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0008 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0009 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0010 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0011 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0012 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0013 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0014 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0015 | 00000000 | 00000100 |          000 |             002 |    DISABLE | 10^-06 |
| 0016 | 00000000 | 00000100 |          000 |             000 |    DISABLE | 10^-06 |
+-----------------------------------------------------------------------------------+
```

Display VC thresholds for SCT 3. Note the following:

The Scaling COSB value applies to congestion in a Class of Service Buffer: if a buffer becomes congested, this value determines how quickly the node throttles back the rate that cells enter the buffer.

The Scaling Log-If is a scaling factor that applies to congestion on a port: when the port is congested, this factor determines the rate at which traffic is throttled back.

```
MGX8850.9.AXSM.a > dspportsct vcThr 3
+-----------------------------------------------------------------------------------------------------------+
Service Class Template [3] : VC Threshold Parameters
+-----------------------------------------------------------------------------------------------------------+
| SERV-TYPE | VC THRESH | PACKET |  MAX_CELL  |   EFCI  | CLP_HI  |  EPD0   | CLP_LO  | SCALING | SCALING |
|           | TBL IDX   |  MODE  |   THRESH   |         |         |         |  EPD1   |  COSB   |  Log-If |
+-----------------------------------------------------------------------------------------------------------+
| VSI-SIG   |    034    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| CBR.1     |    035    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| VBR-RT.1  |    036    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.2  |    037    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-RT.3  |    038    |   DSB  | 0000005000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.1 |    039    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.2 |    040    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| VBR-nRT.3 |    041    |   DSB  | 0000025000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000002 | 0000002 |
| UBR.1     |    042    |   DSB  | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| UBR.2     |    043    |   DSB  | 0000050000 | 1000000 | 0800000 | 0600000 | 0800000 | 0000004 | 0000004 |
| ABR       |    044    |   DSB  | 0000050000 | 0200000 | 0800000 | 0600000 | 0800000 | 0000003 | 0000003 |
| CBR.2     |    045    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
| CBR.3     |    046    |   DSB  | 0000002500 | 1000000 | 0800000 | 0600000 | 0800000 | 0000001 | 0000001 |
+-----------------------------------------------------------------------------------------------------------+
```

Display the Class of Service thresholds for SCT 3.

Note The two *random early discard* parameters (RED Factor and RED Prob) have no application in the current release of the product

```
MGX8850.9.AXSM.a > dspportsct cosThr 3
+-----------------------------------------------------------------------------------------------+
Service Class Template [00003] : COSB Threshold Parameters
+-----------------------------------------------------------------------------------------------+
| COSB |COSB THRESH|  MAX_CELL  |  EFCI   | CLP_HI  |  EPD0   | CLP_LO  |   RED | RED PROB  |
|      | TBL IDX   |   THRESH   |         |         |         |  EPD1   | FACTOR |
+-----------------------------------------------------------------------------------------------+
| 0001 |  0000018  |  1000000   | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0002 |  0000019  |  1000000   | 0200000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0003 |  0000020  |     5000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0004 |  0000021  |    10000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0005 |  0000022  |    50000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0006 |  0000023  |   100000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0007 |  0000024  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0008 |  0000025  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0009 |  0000026  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0010 |  0000027  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0011 |  0000028  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0012 |  0000029  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0013 |  0000030  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0014 |  0000031  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0015 |  0000032  |  1000000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
| 0016 |  0000033  |    10000   | 1000000 | 0800000 | 0600000 | 0800000 | 1000000 | 000000015 |
+-----------------------------------------------------------------------------------------------+
```

```
MGX8850.6.AXSM.a > dspportsct cosb 1
+--------------------------------------------------------------+
|Service Class Template [03] : COSB Parameters                 |
+--------------------------------------------------------------+
|COSB| MIN-RATE| MAX-RATE| EXCESS | CELL DISC |    ERS     |CLR|
|NUM |         |         |PRIORITY| ALARM     |            |   |
+--------------------------------------------------------------+
|  1 |       0 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
|  2 |       6 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
|  3 |       6 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
|  4 |       6 |     100 |     1  | DISABLED  | DISABLED   |  6|
|  5 |       0 |  100000 |     0  | DISABLED  | DISABLED   |  6|
|  6 |       0 |  100000 |     1  | DISABLED  | DISABLED   |  6|
|  7 |       6 |  100000 |     1  | DISABLED  | DISABLED   |  6|
|  8 |       0 |  100000 |     0  | DISABLED  | DISABLED   |  6|
|  9 |       6 |     100 |     1  | DISABLED  | DISABLED   |  6|
| 10 |       0 | 1000000 |     0  | DISABLED  | DISABLED   |  6|
| 11 |       1 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
| 12 |       0 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
| 13 |       0 |  100000 |     2  | DISABLED  | DISABLED   |  6|
| 14 |       0 |  100000 |     2  | DISABLED  | DISABLED   |  6|
| 15 |       6 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
| 16 |       6 | 1000000 |     1  | DISABLED  | DISABLED   |  6|
+------------------------------------------------------------+
```

dspred

Display Redundancy

Displays the current redundant slot links.

Note 1:N redundancy requires a Service Resource Module (SRM) in the switch. The current release does not support SRMs.

Cards on Which This Command Runs

PXM45

Syntax

dspred

Related Commands

addred, delred, switchredcd

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

```
MGX8850.1.AXSM.a > dspred
MGX8850                                              Node Alarm: CRITICAL
Primary  Primary  Primary  Secondary  Secondary  Secondary  Redundancy
SlotNum   Type     State    SlotNum     Type       State       Type
-------  -------  -------  ---------  ---------  ---------  ----------
  7      PXM45     Active      8        PXM45      Empty Resvd  1-1
```

dsprscprtn

Display Resource Partition

Displays information about one resource partition. The displayed information is shown in the example.

Note The **dsppart** and **dsprscprtn** commands are identical. The name 'dsprscprtn' is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Note The connection count includes control VCs when you execute **dsprscprtn** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

The total number of connections in the display includes control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that do not include control VCs, use **dsppnport**.

Cards on Which This Command Runs

AXSM

Syntax

dsprscprtn *<ifNum> <partId>*

Syntax Description

if_num	Logical interface (port) number. For AXSM, the range is 1–60.
part_id	Partition identifier in the range 1–5.

Related Commands

addrscprtn, **cnfrscprtn**, **delrscprtn**, **dsprscprtns**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display configuration for partition 1 on logical port 1 of the current AXSM.

```
MGX8850.1.AXSM.a > dsprscprtn 1 1
  Interface Number              : 1
  Partition Id                  : 1        Number of SPVC: 0
  Controller Id                 : 2        Number of SPVP: 0
  egr Guaranteed bw(.0001percent): 1000000  Number of SVC : 2
  egr Maximum bw(.0001percent)   : 1000000
  ing Guaranteed bw(.0001percent): 1000000
  ing Maximum bw(.0001percent)   : 1000000
  min vpi                       : 0
  max vpi                       : 4095
  min vci                       : 33
  max vci                       : 65535
  guaranteed connections        : 1000
  maximum connections           : 32000 maximum connections            : 4000
```

dsprscprtns

Display Resource Partitions

Display information for all the resource partitions on the current card. The displayed information is shown in the example.

For information on specific elements of a resource partition, see the description of **addrscprtn**.

Note The **dspparts** and **dsprscprtns** commands are identical. The name 'dsprscprtn' is consistent with the corresponding command in Release 1 of the MGX 8850 switch. You can use either command.

Cards on Which This Command Runs

AXSM

Syntax

`dsprscprtns`

Related Commands

addrscprtn, **delrscprtn**, **cnfrscprtn**, **dsprscprtn**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display all resource partitions on the current card.

```
MGX8850.1.AXSM.a > dsprscprtns
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
-----------------------------------------------------------------------------
 1   1    2 1000000 1000000 1000000 1000000    0 4095    33 65535   1000  32000
 2   1    2 1000000 1000000 1000000 1000000    0  255    33 65535   1000  32000
20   1    2 1000000 1000000 1000000 1000000    1    1    33 65535      2    512
21   1    2 1000000 1000000 1000000 1000000    0  255    33 65535      2    512
22   1    2 1000000 1000000 1000000 1000000    0  255    33 65535      2    512
23   1    2 1000000 1000000 1000000 1000000  255  255    33 65535      2    512
```

dspsct

Display SCT

Display the contents of a service class template (SCT) for either the egress or ingress direction. For information on SCTs, see the **cnfcdsct** and **addport** description. Also, refer to the description of SCTs in the *Cisco MGX 8850 Switch Software Configuration Guide*, Release 2.

With the **dspsct** command, you can specify:

- Egress or ingress direction
- A particular SCT template
- A section within the SCT (see Syntax Description for an explanation)

Note Currently, the system does not support certain parameters in the service class templates (SCTs), so you can specify them through **addcon**, **cnfcon**, or Cisco WAN Manager. These parameters are (when applicable) PCR, SCR, and ICR.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dspsct <*SCT_section*> <*sctID*> <egr | ing>

AXSM Syntax Description

SCT_section	A specific part of the SCT, as follows: • **bw** for bandwidth • **gen** for policing and CAC • **cosb** • **vcThr** for VC thresholds • **cosThr** for COSB thresholds
sctID	SCT identifier in the range 1–255.
egr | ing	Specifies the direction to which the template applies. Enter the entire string “eng” or “ing.”

Related Commands

cnfcdsct, **dspcdsct**

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display each parameter in SCT file number 4 in the ingress direction. The output illustrates each category within the SCT file separately.

```
MGX8850.1.AXSM.a > dspsct bw 4 ing
Service Class Template [0] : Bw and Policing Parameters
+-------------------------------------------------------------------------------+
| SERV-TYPE |   PCR    |   SCR    |   MCR    |   MBS    |   CDVT    |   ICR    |
+-------------------------------------------------------------------------------+
| 000000256 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000257 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000258 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000259 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000260 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000261 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000262 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000263 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000264 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000265 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000266 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |
| 000000267 | 00002000 | 00001000 | 00000500 | 00001024 | 00250000 | 00000010 |

MGX8850.6.AXSM.a > dspcdsct gen 4 ing
Service Class Template [0] : General Parameters
+-----------------------------------------------------------+
| SERV-TYPE | COSB_NUM | BOOK_FACT | CAC_TYPE | UPC_ENB  |
+-----------------------------------------------------------+
| 000000256 | 00000003 | 000000100 |   ECAC-A | GCRA1-ENB  |
| 000000257 | 00000004 | 000000100 |   ECAC-A | GCRA 1 & 2 |
| 000000258 | 00000004 | 000000100 |    B-CAC | GCRA 1 & 2 |
| 000000259 | 00000004 | 000000100 |    B-CAC | GCRA 1 & 2 |
| 000000260 | 00000005 | 000000100 |   ECAC-A | GCRA 1 & 2 |
| 000000261 | 00000005 | 000000100 |    B-CAC | GCRA 1 & 2 |
| 000000262 | 00000005 | 000000100 |    B-CAC | GCRA 1 & 2 |
| 000000263 | 00000006 | 000000100 |  LCN_CAC | GCRA1-ENB  |
| 000000264 | 00000006 | 000000100 |  LCN_CAC | GCRA1-ENB  |
| 000000265 | 00000001 | 000000100 |    B-CAC | GCRA1-ENB  |
| 000000266 | 00000003 | 000000100 |    B-CAC | GCRA 1 & 2 |
| 000000267 | 00000003 | 000000100 |    B-CAC | GCRA 1 & 2 |
+-------------------------------------------------------------------------+

MGX8850.6.AXSM.a > dspcdsct cosb 4 ing
+-----------------------------------------------------------------------+
|Service Class Template [00] : COSB Parameters                          |
+-----------------------------------------------------------------------+
| COSB | MIN-RATE | MAX-RATE | MIN-PRIORITY | EXCESS-PRIORITY | ERS ENABLE |
+-----------------------------------------------------------------------+
| 0001 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0002 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0003 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0004 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0005 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0006 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0007 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0008 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0009 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0010 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0011 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0012 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0013 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0014 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0015 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
| 0016 | 00000000 | 00000100 |          001 |             001 |     ENABLE |
```

```
+--------------------------------------------------------------------------+

MGX8850.6.AXSM.a > dspcdsct vcThr 4 ing
Service Class Template [0] : VC Threshold Parameters
+-----------------------------------------------------------------
|SERV|VC THRESH| SELECT |MAX_CELL|EFCI|CLP_LO| CLP_HI| EPD0  |SCALING|SCALING|
|TYPE|TBL IDX  | THRESH | THRESH | Pct|EPDPct|    Pct|   Pct |COSB   | Log-If|
+-----------------------------------------------------------------
| 256 |    225  |   DSB|00000160|100%|035%|080%  |100%  |0000001 |0000001|
| 257 |    226  |   DSB|00001280|100%|035%|080%  |100%  |0000001 |0000001|
| 258 |    227  |   DSB|00001280|100%|035%|080%  |100%  |0000001 |0000001|
| 259 |    228  |   DSB|00001280|100%|035%|080%  |100%  |0000001 |0000001|
| 260 |    229  |   DSB|00001280|100%|080%|080%  |060%  |0000001 |0000001|
| 261 |    230  |   DSB|00001280|100%|080%|080%  |060%  |0000001 |0000001|
| 262 |    231  |   DSB|00001280|100%|080%|080%  |060%  |0000001 |0000001|
| 263 |    232  |   DSB|00008000|100%|080%|080%  |060%  |0000001 |0000001|
| 264 |    233  |   DSB|00008000|100%|080%|080%  |060%  |0000001 |0000001|
| 265 |    234  |   DSB|00008000|020%|080%|080%  |060%  |0000001 |0000001|
| 266 |    235  |   DSB|00000160|100%|035%|080%  |100%  |0000001 |0000001|
| 267 |    236  |   DSB|00000160|100%|035%|080%  |100%  |0000001 |0000001|
+------------------------------------------------------------------------
MGX8850.6.AXSM.a > dspcdsct cosThr 4 ing
+------------------------------------------------------------------------+
|Service Class Template [00] : COSB Threshold Parameters|
+------------------------------------------------------------------------+
| COSB |COSBTHRES| MAX_CELL |EFCI|CLP_LO|CLP_HI| EPD0 |RED  | RED PROB |
|      |TBL IDX  |  THRESH  | Pct|EPDPct|   Pct|  Pct | PCT|  FACTOR  |
+------------------------------------------------------------------------+
| 0001 |  00114  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |\
| 0002 |  00115  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0003 |  00116  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0004 |  00117  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0005 |  00118  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0006 |  00119  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0007 |  00120  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0008 |  00121  | 00001424 |100%|080%  |060%  |100% |025% | 00000005 |
| 0009 |  00122  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0010 |  00123  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0011 |  00124  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0012 |  00125  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0013 |  00126  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0014 |  00127  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0015 |  00128  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
| 0016 |  00129  | 01015808 |100%|100%  |100%  |100% |025% | 00000005 |
+----------------------------------------------------------------------
```

dsptotals

Display Totals

Displays a list of the number current active lines, ports, and channels and the number of maximum lines, ports and channels in the format:

active/maximum

For example, 1/60 indicates that 1 port is active and that the maximum possible ports is 60.

Cards on Which This Command Runs

AXSM

Syntax

dsptotals

Syntax Description

No Parameters.

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

```
MGX8850.2.AXSM.a > dsptotals
total active lines = 2/16
total active ports = 0/60
total active chans = 0/130560
```

switchapsln

Switch APS Line

Switches the specified working APS line (*bay*, *line*) to its protection line.

See the description for the **addapsln** command for a detailed explanation of Automatic Protection Switching (APS).

Cards on Which This Command Runs

AXSM

Syntax

```
switchapsln <bay> <line> <switchOption>
[<serviceSwitch>]
```

Syntax Description

bay	The working bay number to switch.
line	The working line number to switch.
switchOption	The method of performing the switch. 1 = clear (returns to working line) 2 = lockout of protection (locks out the specified APS pair from being switched to protection line) 3 = forced working->protection (forces a working line to protection line switch unless the protection line is locked out) 4 = forced protection->working (forces a protection line to working line switch; 1+1 architecture mode only) 5 = manual working->protection (manual switch) 6 = manual protection->working (manual switch; 1+1 architecture mode only)
service switch	When set to 1, this field causes all APS lines to switch to their protected lines.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, dspapsbkplane, clrbecnt, dspbecnt

Attributes

Log: no log	State: active	Privilege: GROUP1

Example

```
MGX8850.9.AXSM.a > switchapsln 1.1.1 3 1
```

switchredcd

Switch Redundancy Card

Switches operation from the active redundant card to standby redundant card.

Cards on Which This Command Runs

PXM45

Syntax

switchredcd <*fromSlot*> <*toSlot*>

Syntax Description

fromSlot	The currently active redundant card.
toSlot	The current standby redundant card, which is being switched to active.

Related Command

dspred, delred

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

Example

```
mgx8850a.7.PXM.a > switchredcd 8 7
```

uplmi

Up Local Management Interface

Activates the Local Management Interface (LMI) on the specified logical port (*ifNum*).

Cards on Which This Command Runs

AXSM

Syntax

uplmi <*ifNum*>

Syntax Description

ifNum	The interface number of the logical port on which to activate the LMI.

Related Commands

dnlmi

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
uplmi 2
```

upln

Up Line

Activates a line on the current card. After you have activated the line, use **cnfln** to configure the line characteristics such as the type of line (SONET, T3, or E3), line signaling, and so on.

Note See description of **cnfcdsct** for important planning considerations before you use **upln**.

Cards on Which This Command Runs

AXSM

Syntax

```
upln <bay.line>
```

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 3-1.

Related Commands

dspln, **dsplns**, **cnfln**, **dnln**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Activate line 1 in bay 1.

```
MGX8850.1.AXSM.a > upln 1.1
```

upport

Up Port

The **upport** command returns a logical port to the up state (or “ups” the port) so the port can again carry traffic. The **upport** command concludes possible re-configuration or troubleshooting steps. Before you execute **upport**, you must have downed the port by executing **dnport**. Throughout the sequence of downing and upping a port, the configuration for the port remains intact whether the logical port is a UNI or an NNI.

The routes for connections vary by interface type:

- After you re-enable an NNI port through **upport**, you cannot return the re-routed connections to the upped port. The PXM45 routes connections over the trunk as needed.
- On a UNI, the connections continue to exist but remain in the failed state until you enable the port by executing **upport**.

Syntax

```
upport <ifNum>
```

Syntax Description

ifNum	Logical interface (port) number. For AXSM, the range is 1–60. Use **dspports** or **dspport** as needed to determine which port to bring up.

Related Commands

dspport, **dspports**, **dnport**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Restore port 1 on the current card to operation.

```
MGX8850.1.AXSM.a > upport 1
```

CHAPTER 4

ILMI Commands

This chapter describes the ILMI commands. These commands let you add, delete, configure, display status, and create statistics for ILMI at the UNI and for PNNI. The chapter begins with a description of issues related to command entry, port identification, and so on.

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the **Return** key or **Enter** key.

If you press the **Return** key or **Enter** key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide,* Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide,* Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.

- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 4-1.

Table 4-1 Mapping PNNI Port ID to AXSM Elements

PNNI port	**AXSM**
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 4-1 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1
- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

addprfx

Add Prefix—create an address prefix for a UNI or IISP

The **addprfx** command lets you add an ATM prefix for ILMI to a UNI or IISP.

Cards on Which This Command Runs

Syntax

addprfx <*portid*> <*atm-prefix*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2
atm-prefix	A 13-byte ATM prefix (26 hexadecimal characters).

Related Commands

dspprfx, **delprfx**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Add prefix47.0091.8100.0000.0000.0ca7.9e01 to PNNI physical port 3:1.1:1. Display ATM address prefixes for this port. Only the one just added exists.

```
M8850_NY.7.PXM.a > addprfx 3:1.1:1 47.0091.8100.0000.0000.0ca7.9e01

M8850_NY.7.PXM.a > dspprfx 3:1.1:1

ILMI Configured Port Prefix(es):
47.0091.8100.0000.0000.0ca7.9e01

M8850_NY.7.PXM.a >
```

clrilmicnt

Clear ILMI Counters

Clears the ILMI statistics for a partition and logical interface (or port) on a service module.

Cards on Which This Command Runs

AXSM

Syntax

clrilmicnt *<ifNum> <partId>*

Syntax Description

ifNum	Logical port number. On an AXSM, the range is 1–60.
partId	The range for partition identifier is 1–5.

Related Commands

dspilmicnt, dspilmi, dspilmis

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Clear the ILMI statistics for logical interface 1, resource partition 1. Before doing so, confirm the existence of these entities by running the **dspparts** command.

```
pop20two.1.AXSM.a > dspparts
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
----------------------------------------------------------------------------
 1   1    2   10000   10000   10000   10000   10  100   100  1000     0     10


pop20two.1.AXSM.a > clrilmicnt 1 1
ilmi stats for ifNum 1, partId 1 cleared
```

cnfaddrreg

Configure Address Registration

This command lets you enable or disable ILMI address registration for a port. Before you can run **cnfaddrreg**, the following must have occurred:

1. The applicable port must have been created by running the **addpnport** command.
2. The port must be downed by running the **dnpnport** command.

The **cnfaddrreg** command can also enable (or disable) the address registration for backward compatibility.

The peer must support address registration, so you must confirm that address registration is enabled on all three places.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfaddrreg <portid> [{yes | no}]
```

Syntax Description

In addition to typing a port ID, you must also type either “yes” or “no.” The default is “yes.”

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.
yes	Enable ILMI address registration on the port. The default is “yes” (enabled).
no	Disable ILMI address registration on the port.

Related Commands

None

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Disable ILMI address registration on port 4:1.1:11.

```
Geneva.7.PXM45.a > cnfaddrreg 4:1.1:1 no
```

cnfautocnf

Configure Auto Configuration

The **cnfautocnf** command enables or disables ILMI auto configuration for a port. To use this command, the port must be added but administratively down (via **dnpnnport**).

With auto-configuration enabled, the ILMI slave side starts ILMI auto configuration to negotiate the ATM layer parameters with its peer while ports come up. With auto-configuration disabled, the ILMI slave does not start ATM layer parameter-negotiation while ports come up. Instead, the ILMI slave uses the local configuration parameters. The default state for auto-configuration is enabled.

Cards on Which This Command Runs

PXM45

Syntax

cnfautocnf <*portid*> [**yes** | **no**]

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.
yes	Enable ILMI automatic configuration on the port. Default: yes
no	Disable ILMI automatic configuration on the port.

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Enable ILMI auto-configuration on port 4:1.1:11.

```
Geneva.1.AXSM.a > cnfautocnf 4:1.1:1 no
```

cnfilmi

Configure ILMI

The **cnfilmi** command lets you configure the card-level ILMI for the AXSM. Activating the card-level ILMI through **cnfilmi** requires a pre-existing logical port (see **addport**) and resource partition (see **addrscprtn** or **addpart**). No response appears unless an error occurs.

Note For network-level ILMI in relation to PNNI, run the PNNI-specific ILMI commands on the PXM45.

Cards on Which This Command Runs

AXSM

Syntax

```
cnfilmi <ifNum> -id <partitionID> -ilmi <ilmiEnable> -vpi <vpi> -vci <vci> -trap
<ilmiTrapEnable> -s <keepAliveInt> -t <pollingIntervalT491> -k <pollInctFact>
```

Syntax Description

ifNum	Logical port number. On an AXSM, the range is 1–60.
-id	Partition ID in the range 1–5. (See description of **addpart** or **addrscprtn** for information regarding resource partition ID.)
-ilmi	Enable or disable ILMI. 1=enable. 2=disable.
-vpi	VPI for the ILMI signaling connection. The range is 0–255.
-vci	VPI for the ILMI signaling connection. The range is 0–65535.
-trap	Enable or disable ILMI trap. 1=enable. 2=disable.
-s	Keep alive interval. The range is 1–65535 seconds.
-t	Polling interval for T491 in the range 0–65535 seconds.
-k	Polling interval K in the range 0–65535 seconds.

Related Commands

dspilmi, **dspilmis**, **dspilmicnt**, **clrilmicnt**, **dnilmi**, **upilmi**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
Unknown.1.AXSM.a > cnfilmi 1 -id 1 -ilmi 1 -vpi 40 -vci 99 -s 10 -t 10 -k 10
```

cnfilmienable

Enables ILMI on a PNNI port. Prior to **cnfilmienable**, you must use **dnpnport** to de-activate the port.

Cards on Which This Command Runs

PXM45

Syntax

cnfilmienable <*portid*> [**yes** | **no**]

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.

Related Commands

uppnport, **dsppnport**, **dsppnilmi**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Enable ILMI on a PNNI port 1:1.1:2. First, disable the port by using **dnpnport**.

```
M8850_DC.7.PXM.a > dnpnport 1:1.1:2

M8850_DC.7.PXM.a > cnfilmienable 1:1.1:2 yes

M8850_DC.7.PXM.a > uppnport 1:1.1:2

M8850_DC.7.PXM.a > dsppnilmi 1:1.1:2

Port:  1:1.1:2             Port Type:  PNNI             Side:  symmetric
Autoconfig:  enable       UCSM: disable
Secure Link Protocol:  enable
Change of Attachment Point Procedures:  enable
Modification of Local Attributes Standard Procedure:  enable
Addressreg:  Permit All
VPI:       0               VCI:      16
Max Prefix:      16      Total Prefix:       0
Max Address:      64     Total Address:      0
Resync State:      0     Node Prefix: yes
Peer Port Id:    16848898   System_Id : 0.192.0.0.0.0
Peer Addressreg:  enable
Peer Ip Address : 10.10.10.89
Peer Interface Name : atmVirtual.01.1.1.02
ILMI Link State : UpAndNormal
ILMI Version : ilmi30

INFO:  No Prefix registered

INFO:  No ilmi address registered
M8850_DC.7.PXM.a >
```

cnfilmiproto

Configure ILMI Protocol

The **cnfilmiproto** command lets you configure how PNNI reacts to ILMI slave events. Use **dsppnilmi** to confirm changes to the prefix.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfilmiproto <portid> [-securelink {yes|no}] [-attachmentpoint {yes |  no]
[-modlocalattrstd {yes | no]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.
-securelink	Sets the flag **securelink** to make PNNI release the call if it loses connection to the ILMI slave. no: do not enable the ILMI Secure link protocol. yes: disable the ILMI Secure link protocol. Default: yes
-attachmentpoint	Sets the flag **attachmentpoint** to make PNNI release the call if the slave ILMI session sees changes in peer information, such as the system name or system ID. no: do not enable the detection of loss of attachmentpoint protocol. yes: Enable the detection of loss of attachmentpoint. Default: yes
-modlocalattrstd	Sets the flag **modlocalattrstd** to make PNNI release the call if the slave ILMI sees ATM layer (partition resource) changes, such as the VPI or VCI. no: disable the ILMI standard procedure for modification of local ATM param. yes: enable the ILMI standard procedure for modification of local ATM param. Default: yes

Related Commands

dsppnilmi

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

```
SanJose.7.PXM.a > cnfilmiproto 11:2.1.1 -securelink no -attachmentpoint no
-modlocalattrstd yes
```

dbgilmi

Debug ILMI

Use **dbgilmi** to debug ILMI functionality (such as address registration or auto configuration).

Note VSI pass-through information is exchanged between only the controller (PNNI) and the switch.

Cards on Which This Command Runs

PXM45

Syntax

```
dbgilmi {enable | disable}[<portid>]
[-log <vsi | func | minor | major | warning | error | dump | fatal | all>]
[-dbg <vsi | func | minor | major | warning | error | dump | fatal | all>]
```

Syntax Description

enable | **disable**	Activate or de-activate debugging.**dbgilmi**.
portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.
-**log**	vsi func minor major warning error dump fatal all: log all
-**dbg**	vsi func minor major warning error dump fatal all: dbg all

Attributes

Log: log	State: active, standby	Privilege: SERVICE_GP

delprfx

Delete Prefix

The **delprfx** command lets you delete an ILMI address prefix associated with a UN or IISP.

Cards on Which This Command Runs

PXM45

Syntax

delprfx *<portid> <atm-prefix>*

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.
atm-prefix	A 13-byte ATM address prefix, specified as 26 hexadecimal characters.

Related Commands

addprfx, dspprfx

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Delete ATM prefix 47.0091.8100.0000.0000.0ca7.9e01 from PNNI physical port 3:1.1:1. Display prefixes for this port.

```
M8850_NY.7.PXM.a > delprfx 3:1.1:1 47.0091.8100.0000.0000.0ca7.9e01

M8850_NY.7.PXM.a > dspprfx 3:1.1:1

INFO:  No Prefix registered

M8850_NY.7.PXM.a >
```

dnilmi

Down ILMI

The **dnilmi** command lets you de-activate (down) ILMI on a logical port so you can modify a configuration, troubleshoot, or run certain commands that require ILMI to be inoperative.

Cards on Which This Command Runs

AXSM

Syntax

`dnilmi <ifNum> <partId>`

Syntax Description

ifNum	Logical port number. On an AXSM, the range is 1–60.
partId	The range for partition identifier is 1–5.

Related Commands

dspilmi, **dspilmis**, **upilmi**

Attributes

Log: log	State: active, standby	Privilege: SERVICE_GP

dspilmi

Display ILMI

Display the configuration for the interim local management interface (ILMI) on a specific port. The information in the **dspilmi** output was configured through the **cnfilmi** command.

Cards on Which This Command Runs

AXSM

Syntax

dspilmi <*ifNum*> <*partId*>

Syntax Description

ifNum	Logical port number. On an AXSM, the range is 1–60.
partId	The range for partition identifier is 1–5.

Related Commands

cnfilmi, **dspilmis**, **dspilmicnt**, **clrilmicnt**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

This example show the **dspilmi** command line on the AXSM card.

```
pinnacle3.1.2.AXSM.a > dspilmi 1 1

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    1      Off    0   16 On              1          5          4

pinnacle3.2.AXSM.a >
```

dspilmiaddr

Display ILMI Address—displays ILMI registered port addresses.

The **dspilmiaddr** command lets you display the ATM addresses registered by the peer via the ILMI address registration mechanism.

Cards on Which This Command Runs

PXM45

Syntax

`dspilmiaddr` <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.

Related Commands

cnfilmienable, **cnfilmiproto**, **dsppnilmi, pntraceilmi**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the ILMI address on port 3:2.2:4.

```
8850_NY.7.PXM.a > dspilmiaddr 3:2.2:4

INFO:  No ilmi address registered
```

dspilmicnt

Display ILMI Counters

Displays the ILMI counters for a particular resource partition on a particular logical port.

Cards on Which This Command Runs

AXSM

Syntax

dspilmicnt *<ifNum>* *<partId>*

Syntax Description

ifNum	Logical port number. On an AXSM, the range is 1–60.
partId	The range for partition identifier is 1–5.

Related Commands

cnfilmi, **dspilmi**, **dspilmis**, **clrilmicnt**, **dnilmi**, **upilmi**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the ILMI counters for logical port 1/partition ID 1 on the current AXSM card.

```
M8850_NY.1.AXSM.a > dspilmicnt 1 1
If Number                    : 1
Partition Id                 : 1
SNMP Pdu Received            : 1219596
GetRequest Received          : 609802
GetNext Request Received     : 0
SetRequest Received          : 0
Cold Start Trap Received     : 0
GetResponse Received         : 609794
GetResponse Transmitted      : 609802
GetRequest Transmitted       : 609794
Cold Start Trap Transmitted : 1
VPC Trap Transmitted         : 0
VCC Trap Transmitted         : 0
Unknown Type Received        : 0
ASN1 Pdu Parse Error         : 0
No Such Name Error           : 0
Pdu Too Big Error            : 0
```

dspilmis

Display ILMI Configurations

The **dspilmis** command lets you display the configuration of all interim local management interfaces (ILMIs) on the service module.

Cards on Which This Command Runs

AXSM

Syntax

`dspilmis`

Related Commands

cnfilmi, dspilmi, dspilmicnt

Attributes

Log: nolog | State: active, standby | Privilege: ANYUSER

Examples

Display all ILMIs on the current service module.

```
pop20two.1.AXSM.a > dspilmis

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    2    On     0    16    On          1          5          4
    2    2   Off     0    16    On          1          5          4
    3    2   Off     0    16    On          1          5          4

pop20two.1.AXSM.a >
```

dsppnilmi

Display PNNI ILMI

Displays the ILMI information for a PNNI logical port. The ILMI state can be one of the following.

Disable	Protocol is not enabled on this port.
NotApplicable	This port is not accessible due to hardware-related conditions.
LostConnectivity	Protocol on listening port is not enabled.
EnableNotUp	This port is not accessible due to hardware.
UpAndNormal	This port is physically up, and the protocol is enabled.

Note The VC for ILMI is a control channel, but its bandwidth parameters are fixed, as follows: PCR=1000 cps; SCR=50cps; and MBS=1024 cells.

The bandwidth used by ILMI (when enabled) and other control-type VCs (see **cnfpnctlvc**) adds to the bandwidth load on the port. Use **dspload** to determine the load on port resources.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnilmi` `<`*`portid`*`>`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.

Related Commands

dspilmi

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the ILMI for port 6:1.1:1. For comparison, run the **dspilmi** command on the card in slot 6, Note that it contains information that corresponds to the **dsppnilmi** output.

```
M8850_NY.7.PXM.a > dsppnilmi 6:1.1:1

Port:  6:1.1:1             Port Type:  ENNI             Side:  network
Autoconfig:  disable      UCSM: disable
Secure Link Protocol:  enable
Change of Attachment Point Procedures:  enable
Modification of Local Attributes Standard Procedure:  enable
Addressreg:  disable
VPI:       0                VCI:       0
Max Prefix:       0      Total Prefix:       0
Max Address:       0     Total Address:      0
Resync State:      0     Node Prefix: no
Peer Port Id:           0   System_Id : 0.0.0.0.0.0
Peer Addressreg:  disable
Peer Ip Address : 0.0.0.0
Peer Interface Name :
ILMI Link State : Disable
ILMI Version : ilmi40

INFO:  No Prefix registered

INFO:  No ilmi address registered

M8850_NY.7.PXM.a > cc 6

(session redirected)

M8850_NY.6.AXSM.a > dspilmi 1 1

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    1   Off     0    16    On          1          5          4

M8850_NY.6.AXSM.a >
```

dspprfx

Display Prefix

Display the ILMI address prefixes for a port.

Cards on Which This Command Runs

PXM45

Syntax

dspprfx <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 4-2.

Related Commands

addprfx, delprfx

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display all ILMI address prefixes for port 4:1.1:11.

```
SanJose.7.PXM.a > dspprfx 4:1.1:11
ILMI Configured Port Prefix(es):
47.0091.8100.0000.0000.0ca7.9e01
  88.8888.8888.0000.0000.0000.0000

SanJose.7.PXM.a >
```

upilmi

Up ILMI—activate ILMI on a resource partition.

Use **upilmi** to activate interim local management interface (ILMI) for a particular resource partition on a logical port. Before running the **upilmi** command for the partition, you must:

1. Activate a line through the **upln** command and configure the line through **cnfln**
2. Create a logical port through the **addport** command
3. Add resource partitions through **addrscprtn**

After activating ILMI, you can configure ILMI though the **cnfilmi** command.

Cards on Which This Command Runs

AXSM

Syntax

```
upilmi <ifNum> <partId>
```

Syntax Description

ifNum	Logical port number. On the AXSM, the range is 1–60.
partId	The range for partition identifier is 1–5.

Related Commands

cnfilmi, **dspilmi**

Attributes

Log: log	State: active, standby	Privilege: GROUP1

Example

Determine whether ILMI is up on logical port 1, resource partition 1. If ILMI is down (or off), activate it by using the **upilmi** command. Note the second time you run the **dspilmi** command, the ILMI state appears as “on.”

```
M8850_NY.5.AXSME.a > dspilmi 1 1

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    1   Off     0    16   Off          1          5          4

M8850_NY.5.AXSME.a > upilmi 1 1
Warning: connections (if any) on port could get rerouted.
Do you want to proceed (Yes/No) ? y
```

```
M8850_NY.5.AXSM.a > dspilmi 1 1

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    1    On     0    16   Off          1          5          4
```

CHAPTER 5

PNNI Commands

This chapter describes the private network to network (or node to network) interface (PNNI) commands. These commands apply to node addressing and routing on the Cisco MGX 8850 switch, Release 2.0. (For PNNI port and signaling commands, see the chapter, “Logical Node, Port, and Signaling Commands.”) The commands in this chapter have the following applications:

- Configuration
  - Link selection
  - Node addresses and identifiers
  - Packet size
  - PNNI interface
  - PNNI timers
  - RCC variables
  - Routing policies
  - Scope map table
  - PNNI summary address
- Deleting
  - A logical node
  - A PNNI summary address
- Displays of:
  - Debug information
  - Learned nodes
  - Link information
  - Neighbor addresses
  - Node addresses and identifiers
  - Packet information
  - Paths
  - Reachable addresses
  - Routing policy
  - System address tables
  - Timers

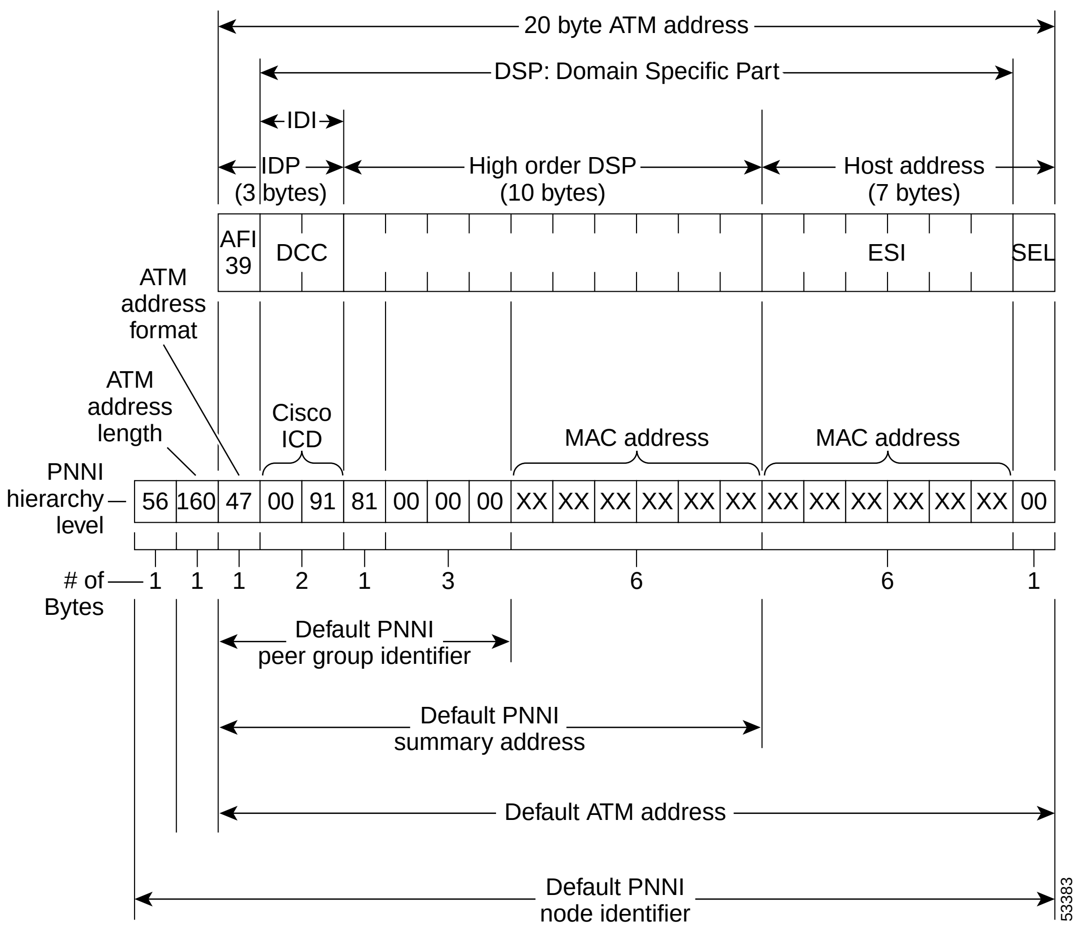
PNNI Addressing

The PNNI addresses and identifiers contain fields that are common to each other and should match each other. If you change a field in one item, Cisco Systems advises you to change the corresponding field in the other item. For example, if you change the ATM address, you should change corresponding fields in the PNNI summary address and node identifier. Note that the peer group identifier does not share in this recommendation.

The name and graphic representation of each address and identifier as well as the common information appears in Figure 5-1. The central and lower parts show the breadth of each item. Applicable command descriptions point out where you should modify common fields. The following fields also go into the construction of the ATM addresses and PNNI-related identifiers:

- The Initial Domain Identifier (IDI) identifies the domain.
- The Initial Domain Part (IDP) identifies the country.
- The Authority and Format Identifier (AFI) identifies the address format and who provided it.
- The Data Country Code (DCC) identifies the country.

Figure 5-1 Cisco Factory-shipped Node-Addressing Defaults Mapping into Both NSAP and DCC Address Formats



Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num.conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 5-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node use the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to the *Cisco MGX 8850 Routing Switch Hardware Installation Guide*, Rel 2.0.

Table 5-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–4	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 5-2.

Table 5-2 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 5-2 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1
- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

addpnni-node

Add PNNI Node—creates a PNNI logical node.

The **addpnni-node** command creates an instance of a PNNI logical node on the switch. Use **addpnni-node** to add a node in the following circumstances:

- After you have removed a node from the topology by executing **delpnni-node**
- After you clear the entire PNNI configuration from the switch by executing **clrallcnf** or **clrcnf**

For a single-peer network, you do not initially need to execute **addpnni-node** because the node comes with a factory-set default. You can use **cnfpnni-node** to modify logical node values as needed.

Note In the current software release, the switch supports only one PNNI logical node—the single peer group setup.

To modify an existing PNNI node, use **cnfpnni-node**. For some of the node parameters, you first must disable the node by executing **cnfpnni-node -enable false**. The following are applicable parameters:

- The *level* is a number in the range 1–104 that shows the relative position of a PNNI logical node within a multi-peer group. For a single peer group, only one level is possible.
- The ATM address applies to the entire switch—only one is necessary for the switch regardless of the release.
- The PNNI node identifier (ID) defines the *logical* node on the switch. As the Syntax Description explains, the node ID consists of numerous fields (see Figure 5-1).
- A peer group identifier defines the nodes in a peer group *within a network*. Logical nodes on more than one physical switch can belong to a peer group.

The sections that follow define the preceding fields. Refer to the description of **cnfpnni-node** for the actual instructions to modify these items. Also, the Syntax Description section for **addpnni-node** provides details about these parameters.

Constraints in the Current Release

For some of the **addpnni-node** parameters, only one value is allowed because the current release supports only single-peer groups. The following list introduces these constraints, and the Syntax Description provides more details.

- You can configure only one PNNI logical node for each switch.
- You must use the defaults for the -**lowest**, -**complexNode**, and -**branchingRestricted** parameters.
- The index for a PNNI logical node on a switch is 1. You cannot configure the index—the system generates it. The index appears in certain display commands, or the CLI prompts you to enter it. (This local index differs from the network-wide index. See **dsppnni-node-list** for details.)

Address Field Descriptions

The **addpnni-node** parameters include two types of identifiers and an ATM address. These identifiers and the ATM address have similar formats and overlapping information. The sections that follow describe these parameters.

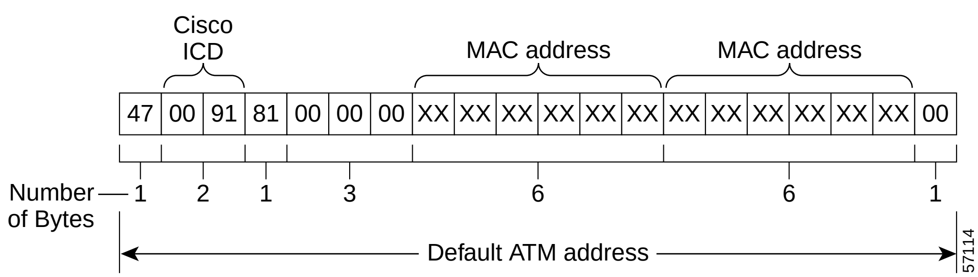
ATM Address of the Node

A switch-level ATM address consists of:

- An eight-bit byte that identifies the format of the address. The format is either E.164 or the more common NSAP.
- 19 8-bit bytes for an ATM address.
- The last byte of the ATM address is the selector byte.
- The Selector byte identifies the host application on the switch. Host applications like PNNI single and multiple peer groups, IP connectivity, and AESA ping have the same ATM address up to the 19 th. byte. The selector byte differentiates between these applications:
  - The selector byte for a single-peer group is 01.
  - The multi-peer group selector bytes are 2–10 and refer to the level of the logical node.
  - AESA ping is 99.

Note For information on port-level ATM addresses, see the description of **addaddr**.

Figure 5-2 Switch-Level ATM Address

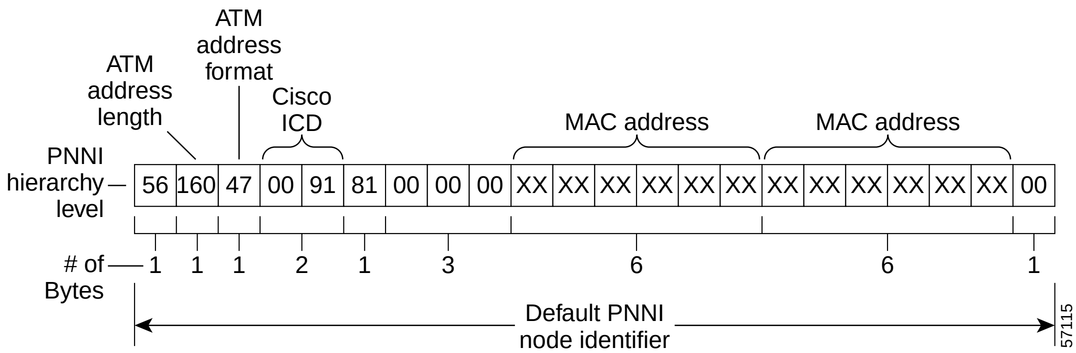


PNNI Logical Node Identifier

A PNNI logical node identifier (node ID) consists of:

- A number that indicates the *level* for the logical node within a hierarchy. The current release supports only one level.
- The *length* of the ATM address for the node ID.
- An ATM address (the same ATM address as the physical switch).
- In Figure 5-3, the *level* is the default of 56. The *length* is 160 bits because the ATM address format is one of the NSAP types—NSAP ICD in this case (as the 47 in the ATM format field shows). For an E.164 address format, the *length* of an ATM address is expressed as decimal digits. For the node-level ATM address, the length is 15. (NSAP address lengths are in bits, whereas E.164 address lengths are in decimal digits.)

Figure 5-3 PNNI Logical Node Identifier

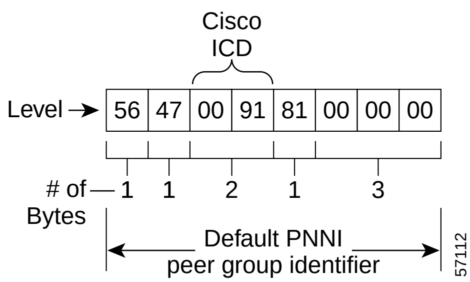


Peer Group Identifier

A peer group identifier (pgID) consists of:

- A *level* that indicates how many bits out of the entire pgID field are actually used.
- A string of hexadecimal bytes copied from the ATM address that uniquely identify the peer group.

Figure 5-4 Peer Group Identifier



Cards on Which This Command Runs

PXM45

Syntax

```
addpnni-node <level>
[lowest | other]]
[-atmAddr atm-address]
[-nodeId node-id]
[-pgId pg-id]
[-enable {true | false}]
[-transitRestricted {on | off}]
[-complexNode{on | off}]
[-branchingRestricted {on | off}]
```

Syntax Description

level	The *level* specifies the level of the node in a PNNI hierarchy and does so by indicating the number of valid bits for a node ID (**-nodeId** parameter) or peer group ID (**-pg-id** parameter). In the current release, you can configure only one level. The level must be the same within the network. The maximum number of levels you can configure on a switch 10. This limit is meaningful only in a multi-peer group. Although the level can any value, selecting an 8-bit boundary makes network planning and address management easier. Four example, using 56 for a level is more expedient than using a level of 59. Range: 1–104 bits Default = 56 bits
lowest | **other**	Indicates whether the logical node you are adding is the lowest node in the hierarchy or exists at a level other than the lowest. Type the entire word "lowest" or "other."In the current release, you can specify only a single-peer group, so any node you add must be the lowest. In this case, this parameter must remain the default of "lowest." If you are adding the node at the lowest level of the switch, you must also specify **-atmAdd** *atm-address*. Default: lowest
-atmAddr	The ATM address of a PNNI logical node consists of 20 hexadecimal, 8-bit bytes (2 hexadecimal characters per byte). If you are adding the lowest node in the switch, you must include an ATM address. For all levels above the lowest level, this ATM address is meaningless. The current release supports only a single peer group, so you must specify an ATM address for the one time that you execute **addpnni-node**. The first byte of *atm-address* indicates the address plan. For example, 47 is reserved for NSAP ICD, as in the following example: 47.00918100000000309409f1f1.00309409f1f1.01 Default: Figure 5-1 shows the Cisco default.
-nodeId	The PNNI logical node identifier (node ID). The *node-id* consists of the following logical elements, starting at the most significant byte: • The level of the PNNI node within the hierarchy. (See the description of the *level* parameter.) • The number of bits in the ATM address. The number is 160 for an NSAP address because the ATM address of the node is always 20 bytes. For an E.164 address, this field is decimal 15. • The ATM address portion of the peer group ID (20 8-bit, hexadecimal bytes—2 hexadecimal characters per byte). Default: The Cisco default appears in Figure 5-1.

-pgId	The peer group ID (*pg-id*) identifies a PNNI peer group. (A PNNI peer group consists of all logical nodes with matching *pg-ids*. In a group of single-peer nodes, each switch has one logical PNNI node.) The number of 8-bit bytes in the peer group ID (*pg-id*) is 14. However, the value of the *level* parameter is the number of bits that are actually used for the peer group ID. For example, with the default *level* of 56 bits (7 bytes), only the first 7 bytes of the peer group ID are relevant. Regardless of the number relevant bytes, the applicable display commands always show 14 bytes for a peer group ID and fill in the irrelevant byte positions with 0s. Default: The Cisco default appears in Figure 5-1.
-enable	Specify the administrative state of the PNNI node. Most applications use the default—node enabled. However, you can add a node in the disabled state according to the necessities of your implementation. For example, you might want to configure the PNNI protocol but not be ready to enable. Later, you can enable it by executing **cnfpnni-node -enable true**. true: Enable this logical node. false: Disable this logical node. Default: true
-transitRestricted	Specifies whether this node can act as a transit node. You can restrict transit to secure the node or to minimize the traffic on a node that either has low-capacity or is high-priority. on: Allow calls to transit this node. off: Allow only call that terminate on this node. Default: off
-complexNode	The current release supports only SPGs, so this parameter currently has no effect. The **complexNode** parameter specifies whether the node is complex. Complex nodes exist only *above* the lowest node in the hierarchy. on: This node is a complex node. off: This node is not a complex node. Default: off
-branchingRestricted	The current release does not support point-to-multipoint branching, so this parameter has no effect. The **branchingRestricted** parameter specifies whether the PNNI node *disallows* point-to-multipoint branches. To enable or disable branching, this parameter actually turns on or off the *restriction*: • To disallow branching, type **-branchingRestricted** on. • To allow branching, type **-branchingRestricted** off. on: Do not allow point-to-multipoint branches. off: Allow point-to-multipoint branches. Default: on

Usage Guidelines

All nodes ship with a default ATM address, node ID, and peer group ID. Cisco uses these defaults to set up and test the switch. Before the switch carries live traffic, you should specify new addresses. For this purpose, you can either use the **cnfpnni-node** command or clear the configuration then use the **addpnni-node** command. (You can clear the PNNI configuration by using either **clrcnf** or **clrallcnf**.) The *Cisco MGX 8850 Routing Switch Software Configuration Guide* explains node-addressing in the section, “Guidelines for Creating an Address Plan.”

Related Commands

cnfpnni-node, delpnni-node, dsppnni-node

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Add a PNNI node with the following configuration then use **dsppnni-node** to check the configuration:

- The PNNI hierarchy level is 56.
- The node is on the lowest level of the PNNI hierarchy.
- The node ATM address is 47.00918100000000309409f1f1.00309409f1f1.01.
- The node PNNI identifier is 56:160:47.00918100000000309409f1f1.00309409f1f1.01.
- The peer group ID is 56:47.009181.0000.00. The number of bytes specified by the *level* parameter is 7 (*level*=56). Therefore, with the *level* itself requiring 1 byte and the *pg-id* having 7 bytes, the total is 8 bytes. As **dsppnni-node** shows, the system adds 0s for the remaining 6 bytes.
- The node is enabled.
- The node permits traffic to cross it on the way to other nodes.

Enter the parameters in a contiguous line. The CLI allows wrapping (not apparent in the example).

```
SanJose.7.PXM.a > addpnni-node 56 -lowest true -atmAddr
47.00918100000000309409f1f1.00309409f1f1.01
-nodeId 56:160:47.0091 81000000 00309409f1f1.00309409f1f1.01 -pgId
56:47.00.9181.0000.0000.0000.0000.00 -enable true -transitRestricted off

SanJose.7.PXM.a >
```

Display the PNNI node configuration.

```
SanJose.7.PXM.a > dsppnni-node

node index: 1                      node name: SanJose
   Level...............        56     Lowest..............      true
   Restricted transit..       off Complex node........       off
   Branching restricted        on
   Admin status........        up     Operational status..        up
   Non-transit for PGL election..       off
   Node id...............56:160:47.00918100000000309409f1f1.00309409f1f1.01
   ATM address...........47.00918100000000309409f1f1.00309409f1f1.01
   Peer group id.........56:47.00.9181.0000.0000.0000.0000.00

SanJose.7.PXM.a >
```

addpnni-summary-addr

Add PNNI Summary Address—create a PNNI summary address to specify a range of addresses.

The **addpnni-summary-addr** command lets you create a summary address for a logical node. The result of this operation is a range of addresses. The controller uses this summary to accept or reject calls that arrive at an address within the address range. Using a summary address decreases both the provisioning time and the call setup time for an address within the range.

Cards on Which This Command Runs

PXM45

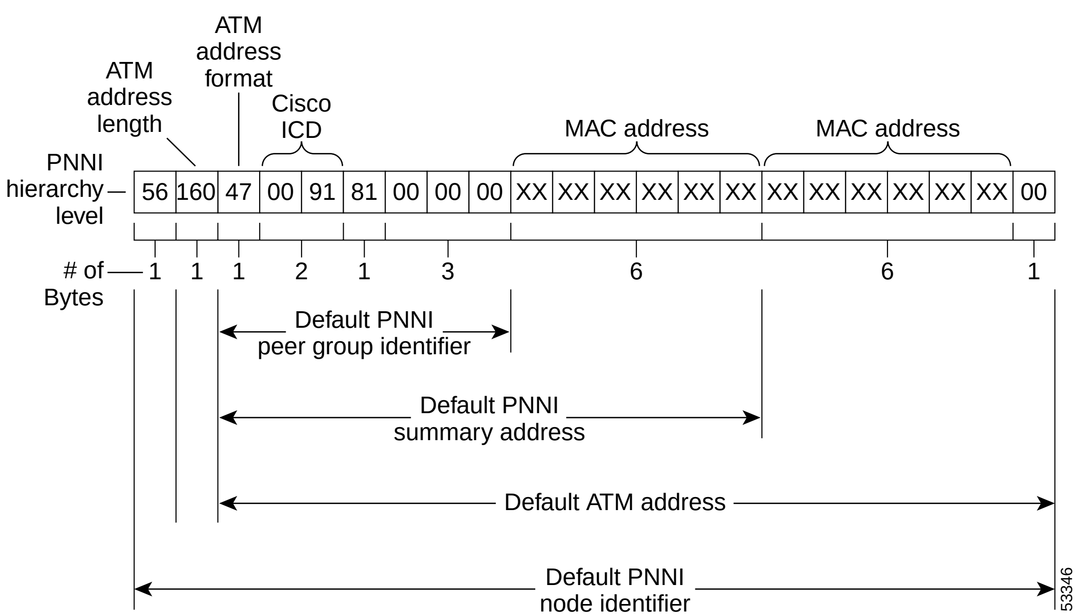
Syntax

```
addpnni-summary-addr <node-index> <address-prefix> <prefix-length>
[-type {internal | exterior}]
[-suppress {true | false}]
```

Syntax Description

The defaults for address prefixes and the peer group ID share values with the ATM address. If you change the peer group ID (**-pgId**) you should change the corresponding fields in the PNNI summary address. (See Figure 5-5.)

Figure 5-5 Cisco Factory-shipped Defaults for PNNI Peer Group Identifier, PNNI Summary Address, ATM Address, and PNNI Node Identifier



node-index	The node index indicates the relative position of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. If you do not have the node index, use **dsppnni-node** to see a list of all logical nodes and node indexes on the current switch. In the current release, the value of *node-index* must be '1'. Range: 1–10 Default: 1
addressprefix	The summary address assigned to the node. The *length* of *addressprefix* is the value of *prefixlength*. As shown in Figure 5-5, *addressprefix* is a formatted hexadecimal string. Default: The default is the first 13 bytes of the *atm-addr*.
prefixlength	Specify the length of the *address-prefix* in bits. The range is 1–152 bits. You configure PNNI routing to look at a specific length in the address, so the length of the PNNI summary address is also configurable. For example, if you configure a node with an 88-bit PNNI summary address, that node sets up a call from any addresses that matches the first 88 bits. The number of addresses that a PNNI summary address can include is inversely related to the length of the PNNI summary address—a shorter summary address can include more addresses than a shorter prefix. In the current release, the zero-length summary address is not supported. Range: 1–152 bits Default: none
-type	Specify the type of the PNNI summary address, either exterior or internal. internal: The summary address includes only addresses that are within the peer group. exterior: The summary address includes addresses that are outside the peer group. Default: internal
-suppress	Specify whether the summary address is advertised to other nodes false: The summary address is advertised (is not suppressed). true: The summary address is not advertised (is suppressed). Default: false

Usage Guidelines

The PNNI summary address table information comes from the internal data base (IDB). If you change or create a PNNI summary address, a topology state packet carries the information to the IDB. The summary address table updates itself from the IDB.

Related Commands

delpnni-summary-addr, dsppnni-summary-addr

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example shows the **addpnni-summary-addr** command line that adds a PNNI address prefix, configured as follows:

- The PNNI summary address is 47.0091.8100.0000.0030.9409.f1f1.
- The length of the PNNI summary address is 104 bits.
- This PNNI summary address contains only internal addresses.
- This PNNI summary address has no advertising suppression (it is advertised).

Use **dsppnni-summary-addr** to display the PNNI address prefixes.

```
SanJose.7.PXM.a > addpnni-summary-addr 1 47.0091.8100.0000.0030.9409.f1f1 104 -type
internal -suppress false
SanJose.7.PXM.a > dsppnni-summary-addr 1

node index: 1
   Type..............    internal     Suppress..............   false
   State............. advertising
   Summary address........47.0091.8100.0000.0030.9409.f1f1/104

SanJose.7.PXM.a >
```

aesa_ping

ATM End Station Address Ping—confirm the connection from a node to an ATM End Station.

The **aesa_ping** command lets you ping any ATM end station address (AESA) connected to a PNNI network. Use this command to check PNNI connectivity to the given destination address. You can use the optional arguments -**setupcall**, -**qos**, -**trace**, and -**data** to send packets and provide greater granularity to the information that the command sends to the screen.

The parameters that you enter in the **aesa_ping** command don't specify anything except the execution of the command itself. All behaviors started by the command stop when the interval -timeout expires.

Cards on Which This Command Runs

PXM45

Syntax

```
aesa_ping <destination address>
[-setupcall {yes | no}]
[-qos {ubr | abr | cbr}]
[-trace {yes | no}]
[-data {enable | disable}]
[-timeout {time out in secs}]
[-interval {time}]
[-pcr {peak cell rate}]
[-scr {sustain cell rate}]
```

Syntax Description

destination address	Set up the destination address in Network Service Access Point (NSAP) format. Default: null
-**setupcall**	Set up a switched virtual connection (SVC) as part of a ping. The call is torn down when the interval expires (see -**timeout** parameter). yes: Set up a call. no: Do not set up a call. Default: no
-qos	The quality of service (QoS) used for switched virtual connection (SVC) ping connection. The QoS can be CBR, ABR, or UBR. Default: UBR

-trace	Specifies whether a path trace is enabled during the ping. The trace stops when the interval specified by **-timeout** elapses. yes: A path trace is enabled. no: No path trace is enabled. Default: no
-data	Specify whether data packets are sent with the ping. The data packets cease when the interval specified by **-timeout** elapses. yes: The data packets are sent. no: No data packets are sent. Default: disable
-timeout	Specify the connection timeout for ping. Any selected ping options also terminate when the interval specified by **-timeout** elapses. Range: 5–120 seconds Default: 5 seconds
-interval	Specify the interval between successive transmissions. Range: 5–120 seconds Default: 5 seconds
-pcr	Specify the peak cell rate. Range: 1–100 cells per second Default: 10
-scr	Specify the sustained cell rate. Range: 1–50 cells per second Default: 5

Related Commands

dsppingatmaddr

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example shows the **aesa_ping** command line that pings the ATM end station with the address 47.00918100000000d058ac23ac.00d058ac23ac.01. The ping is configured as follows:

- No call is set up.
- The QoS metric is UBR.
- No trace is enabled.
- No data is sent with the ping.
- The ping waits six seconds for a reply.
- The ping re-occurs every 60 seconds unless it finishes.
- The peak cell rate of the ping is five cells per second (cps).
- The sustained cell rate of the ping is five cps.

```
SanJose.7.PXM.a > aesa_ping 47.00918100000000d058ac23ac.00d058ac23ac.01 -setupcall no
-qos ubr -trace no -data disable -timeout 6 -interval 60 -pcr 5 -scr 5

Ping Got CLI message, index=0

PING:from PNNI—SOURCE ROUTE
DTL    1 :Number of (Node/port)elements    2

DTL 1:NODE 1::56:160:71:0:145::238:238:238:238:Port 1:262656

DTL 1:NODE 2::56:160:71:0:145::88:172:35:172:Port 2:0

Port List :no of ports =    1

Port ID    1:262656

SanJose.7.PXM.a >
```

cnfpnni-election

Configure PNNI Election—configure a rank and other parameters for electing a peer group leader.

The **cnfpnni-election** command lets you specify the priority of a node for the purpose of electing a peer group leader (PGL). By using this ranking, you can promote or prevent certain nodes for consideration for election as PGL.

Note For a single-peer group, this command has no application.

In a multi-peer group environment, each peer group can elect one PGL. Such an election takes place for every level of the hierarchy. (For example, if three levels exist, three PGL elections occur.) To ensure that a node cannot win the PGL election, you can assign 0 for the priority.

Every node in a peer group runs the election algorithm—with one exception: a node with the Non-Transit for PGL E;ection Flag set.

In addition to the priority, **cnfpnni-election** lets you specify the:

- Number of seconds that the node delays advertising its choice of preferred PGL
- Number of seconds that the node waits to be declared the preferred PGL by unanimous agreement among its peer group members
- Number of seconds that the node waits before it restarts the election of a new PGL after connectivity to the current PGL is lost

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-election <node-index>
[-priority priority]
[-initTime init-time]
[-overrideDelay override-delay]
[-reelectionTime reelection-time]
```

Syntax Description

node-index	The *node-index* uniquely identifies the node within the hierarchy on the switch. Range: 1–10 Default: 1
-priority	Specify the leadership priority within a peer group. Range: 1–20 Default: 0
-initTime	Specify the number of seconds that this node delays advertising its choice of preferred PGL. Default: 15 seconds
-overrideDelay	Specify the number of seconds that this PNNI node waits to be declared the preferred PGL by unanimous agreement among its peer group members. Default: 30 seconds
-reelectionTime	After losing connectivity to the current PGL, specify the number of seconds that this PNNI node waits before restarting the process of electing a new peer group leader. Default: 15 seconds

Related Commands

dsppnni-election

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

In the current release, this command has no meaning.

cnfpnni-intf

Configure PNNI Interface—specify administrative weight or logical link aggregation for a port.

The **cnfpnni-intf** command lets you specify two distinct PNNI parameters. The *aggregation token* applies to a logical node, and the *administrative weight* (AW) applies to a port.

> **Note** Aggregation tokens do not apply in a single-peer group.

Aggregation Tokens

An aggregation token is a marker that indicates which up-links (links going out of the peer group) can be bundled or aggregated at the next logical level. For example, if four links *a*, *b*, *c*, and *d* connect to peer groups X and Y, no aggregation exists for the links by default. The LGNs of these peer groups could have hypothetical names x02 and y02. They would be connected by a single logical link. You could use the **cnfpnni-intf** command to configure aggregation tokens.

If you configure the aggregation token of link c as 1, then x02 and y02 would have two logical links. One link would correspond to aggregation token 0, and the other link would correspond to aggregation token 1. In summary:

> An aggregation token is the number of links between a peer group leader and the lowest level of a PNNI hierarchy. The range for tokens is 1–32.

Administrative Weight

The administrative weight (AW) is a number that serves as a cost-based determinant of a route. Each port in a PNNI network has a default AW for the egress direction. Whether it uses the default AW or an AW you specify, PNNI adds all the AWs in a prospective route then determines whether the route is too expensive. For an SPVC, you can specify the maximum cost with the **addcon** or **cnfcon** command.

You can specify that *all* classes of service on the port have the same AW, or you can specify an AW for *each service class* on the port. If you specify the same AW for *all*, it overrides the default AW or the AW you might specify for individual services types.

The AW for a path is the sum, in both directions, of the individual weight of each link on the path. See “Usage Guidelines” section on page 23 for details about AW.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-intf <portid>
[-aggregationToken token]
[-awcbr aw]
[-awrtvbr aw]
[-awnrtvbr aw]
[-awabr aw]
[-awubr aw]
[-awall aw]
```

Syntax Description

If you enter **-awall** to specify an AW, it overrides the AW for all individual service types.

t

portid	The PNNI physical port ID. The *portid* has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. For more details about *portid*, see PNNI Format. Default: none
-aggregationToken	Specify a 32-bit number for logical link aggregation between a peer group leader and the lowest level. Range: 0–2147483648 Default: 0 The current release does not support multiple PNNI levels, so this parameter has no meaning.
-awcbr	Specify the administrative weight for constant bit rate (CBR) on this interface. Range: 0–4194304 Default: 5040
-awrtvbr	Specify the administrative weight for real-time variable bit rate (rt-VBR) on this interface. Range: 0–4194304 Default: 5040
-awnrtvbr	Specify the administrative weight for non-real-time variable bit rate (nrt-VBR) on this interface. Range: 0–4194304 Default: 5040
-awabr	Specify the administrative weight for available bit rate (ABR) on this interface. Range: 0–4194304 Default: 5040
-awubr	Specify the administrative weight for unspecified bit rate (UBR) on this interface. UBR connections can include SVC ping connections. Range: 0–4194304 Default: 5040
-awall	Specify the administrative weight for all service types on this interface. If you use **-awall**, this *aw* becomes the AW for every service type on this port. Range: 0–4194304 Default: 5040

Usage Guidelines

PNNI includes a protocol for routing the topology state. This protocol advertises details about the peer group links and nodes. Links and nodes are assigned attributes that can be used to diagnose and tune network behavior.

The administrative weight (AW) for a port is an integer that has no units of measure. The switch compares the sum of all AWs along a path with the sum of all AWs along another path to determine which path is more cost-effective. You can specify one AW for all classes of service, or you can specify an AW for each class of service. The AW for all (-**awall**) overrides the AW for an individual class.

The AW parameter influences how PNNI selects a path within a peer group and therefore how it distributes SVCs and SPVCs. PNNI route selection can also use AW to exclude certain links from routing. For example, it can define a backup link for use only when insufficient bandwidth is available on the primary link.

Related Commands

dsppnni-intf

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Specify the following AWs on port 4:1.1:11.

- The AW for real time variable bit rate is 11040.
- The AW for non-real time variable bit rate is 20040.
- The AW for unspecified bit rate is 1040.

Use the **dsppnni-intf** command to display the configuration.

```
SanJose.7.PXM.a > cnfpnni-intf 4:1.1:11 -awrtvbr 11040 -awnrtvbr 20040 -awubr 1040

SanJose.7.PXM.a > dsppnni-intf 4:1.1:11

Physical port id: 4: 1.1:11        Logical port id:   17045515
   Aggr token..........         0     AW-NRTVBR...........     20040
   AW-CBR..............      5040     AW-ABR..............     5040
   AW-RTVBR............     11040     AW-UBR..............      1040

SanJose.7.PXM.a >
```

cnfpnni-link-selection

Configure PNNI Link Selection—specify the routing policies for parallel links.

The **cnfpnni-link-selection** command specifies which routing policies are used to select one of the parallel links that connect a neighboring PNNI node. Neighboring node in this case means a directly connected node. The **cnfpnni-link-selection** command applies only if parallel links exist between the specified port and neighboring nodes.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-link-selection <portid>
{minaw | maxavcr | maxcr | loadbalance}
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 5-5. Default: no default port ID
minaw	The routing protocol selects the link with the least administrative weight (AW). The AW is a numeric value that every logical port ingress has. The total cost of a route is the sum of all ingress port AWs in both directions of the route. The minaw policy is the default.
maxavcr	The routing protocol selects the link with the largest available cell rate (AvCR). The AvCR is the remaining bandwidth after the total utilized bandwidth is subtracted from the maximum cell rate (MaxCR). The maxavcr policy works well for traffic that consistently requires a large amount of bandwidth.
maxcr	The routing protocol selects the link with the largest maximum cell rate (MaxCR). Typically, the MaxCR is the maximum speed of the line (or trunk). Therefore, if one link traverses an OC3 line and another link traverses a T3 line, the MaxCR policy dictates that PNNI select the OC3 line. Only when sufficient bandwidth does not exist on a line does the routing protocol switch to AvCR. The maxcr policy works well for bursty traffic.
loadbalance	The routing protocol alternates the link it selects for routing new calls. The load balance policy works well with links that have identical or very similar AW or bandwidth characteristics.

Related Commands

dsppnni-link-selection

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Specify maximum available cell rate as the routing policy for the link with port ID 4:1.1:11. Use **dsppnni-link-selection** to see the link selection policy on this port. Note that **dsppnni-link-selection** shows the logical port number for the physical port ID.

```
SanJose.7.PXM.a > cnfpnni-link-selection 4:1.1:11 maxavcr
SanJose.7.PXM.a > dsppnni-link-selection 4:1.1:11

physical port id:        4:1.1:11     link selection: maxavcr
 logical port id:        17045515


SanJose.7.PXM.a >
```

cnfpnni-mtu

Configure PNNI Maximum Transfer Unit—specify the maximum data packet size that the node supports.

The **cnfpnni-mtu** command specifies the maximum data packet size in number of bytes. This command is useful for internetworking compatibility, for which you must match the maximum packet size of your peer group to a size that another peer group can support. Although this command applies primarily to internetwork compatibility, you can also use it in trials to test the affect of various packet sizes on the performance of the peer group.

Use **dsppnni-mtu** to display the PNNI packet size configuration.

Cards on Which This Command Runs

PXM45

Syntax

cnfpnni-mtu <*mtu*>

Syntax Description

mtu	Specify maximum transmit unit in number of bytes. Range: 2048–8192 bytes Default: 2048 bytes

Related Commands

dsppnni-mtu

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example shows the **cnfpnni-mtu** command line that specifies the maximum PNNI packet size to 3002 bytes. Use **dsppnni-mtu** to display the specified information.

```
SanJose.7.PXM.a > cnfpnni-mtu 3002

SanJose.7.PXM.a > dsppnni-mtu

max packet size: 3002

SanJose.7.PXM.a >
```

cnfpnni-node

Configure PNNI Node—modify parameters for an existing PNNI node.

The node-level **cnfpnni-node** command lets you:

- Modify the parameters of an existing logical node.
- Enable or disable a node. This ability applies to:
  - Parameters that require the node to be disabled before you can modify them (if you do not modify parameters that require a disabled node, you can modify one or more parameters with one execution of **cnfpnni-node**).
  - A situation where you created the node in the disabled state with **addpnni-node.**

The node configuration and state parameters you can modify are as follows:

- The ATM address—requires the node to be disabled
- The level of the node within the hierarchy—requires the node to be disabled
- The node ID—requires the node to be disabled
- The peer group ID—requires the node to be disabled
- Whether the node is the lowest node in the hierarchy
- The state of the node—enabled or disabled
- Whether connections can transit this node
- Whether or not the node is complex
- Whether the node supports point-to-multipoint connections

The preceding parameters appear as optional parameters in the syntax. The only required parameter for this command is the local *node index*. In the current release, the only node index is "1."

Note When this command runs, existing calls are not affected. Modified parameters apply to new routing after the command finishes.

Disabling a Node When Required

For some of parameters, you must *first* disable the node. Applicable parameters are the:

- Level
- ATM address
- Node ID
- Peer group ID

For these parameters, three executions of **cnfpnni-node** are necessary:

1. **cnfpnni-node -enable** false
2. **cnfpnni-node** (*change parameters*, including those that do not require a disabled node)
3. **cnfpnni-node -enable** true

Be sure to synchronize applicable changes (such a change in level) throughout the network and follow up with **dsppnni-node** or other applicable display commands.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-node <node-index>
[-atmAddr atm-address]
[-level level]
[-nodeId node-id]
[-pgId pg-id]
[-lowest {true | false}]
[-enable {true | false}]
[-transitRestricted {on | off}]
[-complexNode{on | off}]
[-branchingRestricted {on | off}]
```

Syntax Description

node-index	The node index specifies the relative position of a logical node within the hierarchy of a multi-peer group. The lowest level is 1. In the current release, the only *node-index* is ‘1’. Each new logical node added to the hierarchy automatically gets the next higher index number, so you cannot configure the node index. Range: 1–10 Default: 1
-atmAddr	Specify the ATM address for this logical node. For you to change the ATM address, the node must be disabled. For details, see the section, “Disabling a Node When Required.” Note that only the lowest node in the hierarchy requires an ATM address. Default: Figure 5-6 shows the factory-set default.
-**level**	This parameter specifies the level of the node within the PNNI hierarchy. The level of the node is the number of bits in the node ID (-**nodeId** parameter) or peer group ID (-**pg-id** parameter). For example, the default level of 56 means that the node ID is 56 bits long. If you specify a level of 48, the node ID has a length of 48 bits. The maximum number of levels you can configure on a switch 10. This limit is meaningful in a multi-peer group only. Although the level can be any value within the 1–104 range, selecting an 8-bit boundary makes network planning and address management easier. Four example, using 56 for a level is more expedient than using a level of 59. Range: 1–104 bits Default: 56 bits

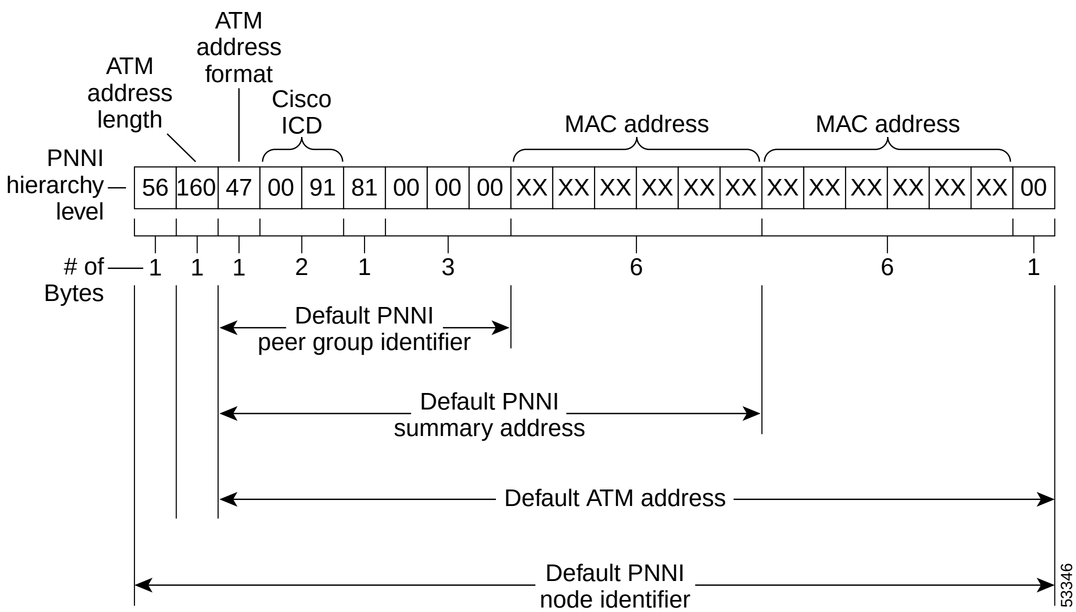
-nodeId	Specify the PNNI node identifier assigned to a PNNI node. The *node-id* consists of the PNNI hierarchy level (**-level)** followed by the length of the ATM address (**addaddr** *length)* followed by the ATM address (**-atmAddr)**. **Note** Before you change the node ID, you must first use the **cnfpnni-node** command one time with the parameter string **-enable false** to disable the node. After changing the node ID, run the command with the parameter string -**enable true.** As shown in Figure 5-6, -**nodeId** is a 22-byte, formatted hexadecimal string. Like all PNNI addresses, identifiers, and prefixes, this value is portrayed as a string of hexadecimal "nibbles." One or several pairs of nibbles entail each parameter field. Default: Figure 5-6 shows the factory-set default.
-pgId	All members of a peer group have the same peer group identifier (pgID) and exist on the same level. (The level is either the existing number of bits or whatever you specify with -**level** *level*). The default value of *level* is 56 (7 bytes), which specifies the length of **-pgId** to 7 bytes. However, the maximum length of **-pgId** is 14 bytes, so display commands always show -**pgId** as 14 bytes with trailing zeros filling the undefined fields. If you increase the value of *level*, you change the length, and therefore the value, of **-pgId**, but it will always be displayed as 14 bytes. Before you change the value of *pg-id*, disable the node by entering **cnfpnni-node -enable false**. This is a 14-byte, formatted hexadecimal string. Default: Figure 5-6 shows the factory-set default.
-enable	Specify the administrative status of the PNNI node. Before you change a node ATM address, node ID, peer group ID, or hierarchical level, the node must be disabled. true: Enable the node. false: Disable the node. Default: true
-transitRestricted	Specify whether connections can transit this node. You can disallow *via* connections for security reasons, to minimize traffic supported by either a low bandwidth node or a highly critical node, and so on. on: This node allows via connections. off: Via connections cannot transit this node. Default: off

-complexNode	Specify whether this node is a complex node. The lowest-level node cannot be a complex node. on: This node is a complex node. off: This node is not a complex node. Default: off **Note** This release does not support multiple peer groups, so this parameter must remain off.
-branchingRestricted	Specify whether the PNNI node allows point-to-multipoint branches. on: This node does not allow point-to-multipoint branches. off: This node allows point-to-multipoint branches. Default: on **Note** This release does not support branching, so this parameter must remain on.

Note In the current release, you must keep the default values for -**lowest**, -**complexNode**, and -**branchingRestricted**.

Caution Cisco factory-set defaults for address prefixes and the peer group ID share field values with the ATM address. If you change the peer group ID, you should change the corresponding fields in the ATM address and node identifier. (See Figure 5-6 for the mapping between these addresses.)

Figure 5-6 Cisco Factory-shipped Defaults for PNNI Peer Group Identifier, PNNI Summary Address, ATM Address, and PNNI Node Identifier



Usage Guidelines

All MGX 8850 cards ship with default addresses. These defaults are provided for evaluation of the switch. Before or while you deploy a switch, you should change the default addresses by executing either **cnfpnni-node** or **addpnni-node**. The *Cisco MGX 8850 Routing Switch Software Configuration Guide* explains node addressing in the section Guidelines for Creating an Address Plan.

The ATM address, address prefixes, and peer group ID share some default field values, as shown in Figure 5-6.

Related Commands

addpnni-node, **delpnni-node**, **dsppnni-node**, **addaddr**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example shows the **cnfpnni-node** command line that specifies the PNNI configuration values for a node that already exists on the network. The command line configures the PNNI node as follows:

- The PNNI hierarchy level is 56.
- The node ATM address is 47.0091 8100000000309409f1f1.00309409f1f1.01.
- The node PNNI identifier is 56:160:47.00918100000000309409f1f1.00309409f1f1.01.
- The node PNNI group identifier is 56:47.009181.0000.00. The length specified by the hierarchy level is 56 (8 bytes) the following 12 zeros are fill.

- The node is at the lowest level of the network PNNI hierarchy.
- The node is enabled.
- The node does not permit traffic to cross it on the way to other nodes.
- The node is not a complex node.
- The node permits branching.

Use **dsppnni-node** to display the PNNI configuration values.

```
SanJose.7.PXM.a > cnfpnni-node 1 -level 56 -nodeId 56:160:47.0091 81000000 00309409f1f1.00309409f1f1.01
-atmAddr 47.0091 81000000 00309409f1f1.00309409f1f1.01 -pgId 56:47.00.9181.0000.0000.0000.0000.00
-lowest true -enable true -transitRestricted on -complexNode off -branchingRestricted off

SanJose.7.PXM.a > dsppnni-node

node index: 1                      node name: SanJose
   Level...............        56     Lowest..............      true
   Restricted transit..       on      Complex node........       off
   Branching restricted        off
   Admin status........        up     Operational status..        up
   Non-transit for PGL election..       off
   Node id...............56:160:47.00918100000000309409f1f1.00309409f1f1.01
   ATM address...........47.00918100000000309409f1f1.00309409f1f1.01
   Peer group id.........56:47.00.9181.0000.0000.0000.0000.00

SanJose.7.PXM.a >
```

cnfpnni-pkttrace

Configure PNNI Packet Trace—specify the parameters to trace packets.

This command applies to debugging. It causes the switch to check the PNNI packets transmitted and received at each peer. For example, if an interface is not receiving user-traffic, you can execute **dsppnni-pktrace** to display packet information that can help you find the level and type of blockage.

The **cnfpnni-pkttrace** command specifies the trace command examines the contents of PNNI packets exchanged between two neighboring nodes. A PNNI packet consists of a Hello packet and the topology information packets.

Note Executing **cnfpnni-pkttrace** can significantly increase operational overhead, thereby degrading network performance. You should consider executing this command while the node carries little or no live traffic. If you execute **cnfpnni-pkttrace** while live traffic is present, you may want to consider tracing the packets for one direction at a time (transmit or receive).

If you plan to use the optional **-portId** parameter but do not have the logical format of the port ID, use **dsppnport**. For details, refer to the Syntax Description of **cnfpnni-pkttrace**.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-pkttrace {tx {on|off} | rx {on|off} } [node-index [-portId port-id |
-svcIndex svc-index]]
```

Syntax Description

tx	Specify whether the switch traces transmitted packets. on: Trace the transmitted packets off: Do not trace the transmitted packets Default: (no default)
rx	Specify whether the switch traces received packets. on: Trace the received packets off: Do not trace the received packets Default = (no default)
node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default = 1

-portId The port ID in this instance has the format of the logical ID number. The format is a 32-bit encoded number in the range 1–2147483648. If you do not have the port ID in this form, use **dsppnport** and provide it with the common portID format of *slot*[:*subslot*].*port*[:*subport*}. The output of **dsppnport** shows the logical number for the port ID. Use this value is for the -**portID** parameter.

-**svcIndex** PNNI uses the SVC index as a reference to the SVC-based, logical, horizontal link. This parameter is meaningful only if you specify node-index.

Default: none

Note The current release does not support Routing Control Channels for Switched Virtual Connections (SVCC-RCC), so this value must remain 0.

Usage Guidelines

PNNI trace commands have characteristics that standard debug commands lack, namely:

- Trace commands debug interactions between different software modules or within a module.
- Trace output goes to a system trace buffer, not to the console.
- Tracing controls a more granular filtering of unnecessary debug output.

Related Commands

dsppnni-pkttrace, **dsppnport**, **dsppnports**

Attributes

Log: log State: active Privilege: SERVICE_GP

Example

Configure then display a trace that examines the contents of PNNI Hello packets, as follows:

- The *transmit* packets are traced.
- The packet trace occurs at node index of 1 (the default for **cnfpnni-pkttrace** and therefore omitted).
- The packet trace takes place on the port identifier of 17504.

```
Geneva.7.PXM.a > cnfpnni-pkttrace tx 17504
PNNI/tx_packet on port 17504 at level 56
> 01:00010064  01010100  000038a0  47009181  00000000  309409f3  b8003094
> 02:09f3b801  47009181  00000000  309409f3  b8003094  09f3b801  38470091
> 03:81000000  00000000  000038a0  47009181  00000000  001a531c  2a00001a

Geneva.7.PXM.a > dsppnni-pkttrace tx -portId 17504

Node Index :1   Port id:        17504   Tx Pkt Trace on

Geneva.7.PXM.a >
```

cnfpnni-routing-policy

Configure PNNI Routing Policy—specify the routing policy parameters.

The **cnfpnni-routing-policy** command specifies which routing policy values are used during UNI call setup. Routing policies control PNNI routing for your network.

The **cnfpnni-routing-policy** command specifies which routing policies are used for:

- Generating background routing tables
- Load balancing
- On-demand routing—applies to crankback or a situation where the node must route a call to a destination for which no route exists in the pre-computed routing tables

The two dynamic routing protocols are *shortest path* and *on-demand* routing.

Background Routing Tables

The background routing tables contain all routes within the peer group. The routes are calculated from information in the internal data base (IDB). When a topology change affects usable routes within the peer group, a PNNI topology state packet carries the formation to each node so it can update its IDB. The background routing tables are updated to reflect the change in routing parameters.

Routing tables are generated (or pre-computed) to support routing based on the shortest path.

- The administrative weight (AW) background routing table stores the AWs for all routes.
- The cell transfer delay (CTD) background routing table stores the CTD data for known routes.
- The cell delay variation (CDV) background routing table stores the CDV data for known routes.

On-demand routing is used if any of the following occur:

- All pre-computed routing tables are disabled.
- A route does not appear in the precomputed routing tables.
- Crankback is enabled within a designated transit list (DTL).

Caution Poor routing policies can cripple or even crash a network. You should not change routing policies on a deployed network unless you have carefully planned the changes and know how the changes can affect the network.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-routing-policy
[-sptEpsilon epsilon]
[-sptHolddown holddown]
[-bnPathHolddown bn-path-holddown]
[-loadBalance {random | maxbw}]
[-onDemand {firstfit | bestfit}]
[-awBgTable {on|off}]
[-ctdBgTable {on|off}]
[-cdvBgTable {on|off}]
```

Syntax Description

-sptEpsilon

The shortest path (SPT) *epsilon* you supply specifies a tolerance in the form of a percent that can influence which paths qualify as equal-cost during route calculation. A higher tolerance results in a broader range of path cost-values that can qualify as equal-cost. If two paths have very similar administrative weights (AWs), a large enough tolerance eliminates equal-cost as a routing factor because the routing algorithm regards the costs as equal.

The range of 0–20 for this parameter comes from the ATM Forum PNNI specification. However, the percent of tolerance that the numbers dictate is determined by individual vendors. Cisco Systems currently maps the following percentages for the Cisco MGX 8850 switch:

0: the total AWs along both directions of the route must be identical.
1-2: the total AWs along both directions of the route must be within 1.06%.
3-4: the total AWs along both directions of the route must be within 3.125%.
5-9: the total AWs along both directions of the route must be within 6.25%.
10-15: the total AWs along both directions of the route must be within 12.5%.
16-20: the total AWs along both directions of the route must be within 25.0%.

Range: 0–20
Default: 0, which means the cost of two paths must be *identical* to qualify as equal-cost

-sptHolddown

The interval between two consecutive calculations for generating routing tables. If a network is stable, it may not be necessary to generate routing tables 10 times per second. In such a case, you can increase the value to reclaim CPU time needlessly used to update unchanging routing tables.

Units: 100 millisecond increments
Range: 1–600 (0.1–60 seconds)
Default: 1

-bnPathHolddown

The minimum interval between consecutive calculations of routing tables for border nodes. For a stable network, generating the routing tables 10 times a second may be unnecessary. If this case, you can increase the interval to save the CPU time spent on updating routing tables that are not changing.

Note The current release does not support complex nodes, so leave this parameter as is.

Units: 100 milliseconds
Range: 2–600 (0.2–60 seconds)
Default: 2

-loadBalance	The PNNI routing protocol relies on the load balance policy when it determines that the routes to a destination have equal cost. (PNNI determines routes to be identical according to the metrics in the AW, CDV, or CTD routing table.) Note that the SPT epsilon value can be a factor when PNNI calculates cost. The choice for load balancing is “random” or “maxbw.” Type the entire word. With “random” load balancing, PNNI randomly chooses between the equal cost routes. This rule requires less computational overhead. With “maxbw” load balancing, PNNI selects the route with more available bandwidth when it chooses between equal cost routes. This rule has more overhead due to ongoing comparison of available bandwidth on the routes. Default: no default
-onDemand	On-demand routing applies to crankback or a situation where the node must route a call to a destination for which no route exists in the pre-computed routing tables. The on-demand policy is either “firstfit” and “bestfit.” With firstfit (the default), PNNI selects the first route to the destination. This approach minimizes search time but may not result in the best route. With bestfit, PNNI selects a route based on: • The route with the lowest cost (see AW in the **cnfpnni-intf** description and maxcost in the **addcon** description because these parameters are closely related). • Link verification. • Path constraint checks. • Avoidance of blocked nodes and links. • Checking limits in the designated transit list (DTL). With bestfit, PNNI selects the optimum route but entails greater computing overhead. The search-time depends on the density and complexity of the network. Default: firstfit
-awBgTable	Enable or disable the background routing table for AW. on: The background routing table of AW is enabled. off: The background routing table of AW is disabled. Default: on

-ctdBgTable	Enable or disable the background routing table of CTD. The CTD parameter is the interval between a cell exiting the source PNNI node and entering the destination PNNI node. on: The background routing table of CTD is enabled. off: The background routing table of CTD is disabled. Default = on
-cdvBgTable	Enable or disable the background routing table of CDV. The CDV parameter is a component of cell transfer delay, and is a quality of service (QoS) delay parameter associated with CBR and VBR service. Cell Delay Variation is the variation of delay between cells, measured peak to peak. on: The background routing table of CDV is enabled. off: The background routing table of CDV is disabled. Default = on

Related Commands

dsppnni-routing-policy

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Specify the following routing policy:

- Only paths with identical values qualify as equal-cost paths.
- The routing tables are generated every 0.1 seconds.
- The load balancing is purely random.
- On-demand routing selects to the optimal route.
- The background routing table for AW is enabled.
- The background routing table for cell transfer delay is enabled.
- The background routing table for cell delay variation is enabled.

You can confirm your settings with the related command **dsppnni-routing-policy**.

```
Geneva.7.PXM.a > cnfpnni-routing-policy -sptEpsilon 0 -sptHolddown 1 -loadBalance random
-onDemand bestfit -awBgTable on -ctdBgTable on -cdvBgTable on

Geneva.7.PXM.a > dsppnni-routing-policy

SPT epsilon......... 0 Load balance........ random
SPT holddown time... 1 On demand routing... best fit
SPT path holddown time 2 AW Background Table on
CTD Background Table on CDV Background Table on
Geneva.7.PXM.a >
```

cnfpnni-scope-map

Configure PNNI Scope Map Table—specify the table that maps the UNI 4.0 values to the PNNI hierarchy levels.

The **cnfpnni-scope-map** command specifies how UNI 4.0 address scope values are mapped to PNNI hierarchal levels. These parameters are stored in the scope map table.

Note This function requires support of UNI 4.0 and therefore is not supported in this release.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-scope-map <scope> <level>
```

Syntax Description

scope	Specify the UNI 4.0 address scope. Range: 1–15, where: 1 = LocalNetwork 2 = LocalNetworkPlusOne 3 = LocalNetworkPlusTwo 4 = SiteMinusOne 5 = IntraSite 6 = SitePlusOne 7 = OrganizationMinusOne 8 = IntraOrganization 9 = OrganizationPlusOne 10 = CommunityMinusOne 11 = IntraCommunity 12 = CommunityPlusOne 13 = Regional 14 = InterRegional 15 = Global Default: none
level	Specify the PNNI level to which the UNI 4.0 address scope is mapped. Range: 1–104 Default: 56

Related Commands

dsppnni-scope-map

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This command is not supported by the current release.

cnfpnni-svcc-rcc-timer

Configure PNNI Switched Virtual Connection Routing Control Channel (SVCC-RCC) Timer—specify the parameters for any PNNI RCCs between logical nodes.

The current release does not support this command. The **cnfpnni-svcc-rcc-timer** command lets you specify the initial PNNI SVCC-based variables for a network node. Logical group nodes (LGNs) use SVCC-RCCs to exchange routing information.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-svcc-rcc-timer <node-index>
[-initTime init-time]
[-retryTime retry-time]
[-callingIntegrityTime calling-integrity-time]
[-calledIntegrityTime called-integrity-time]
```

Syntax Description

node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default = 1
-initTime	The number of seconds this node waits before it advertises its choice of preferred SVCC to a neighbor with a numerically lower ATM address. This node does so after determining that such an SVCC should be established. Range: 1–10 seconds Default = 10
-retryTime	The number of seconds this node waits before it attempts to re-establish an apparently necessary and viable SVCC-based RCC that was unexpectedly torn down. Range: 10–60 seconds Default = 60

-callingIntegrityTime	The number of seconds the node waits while it attempts to set up an SVCC as the calling party. If the SVCC is not fully established within this time period, the node tears down the connection. Range: 5–300. Default = 300.
-calledIntegrityTime	The number of seconds the node waits while it attempts to set up an SVCC as the called party. If the SVCC is not fully established within this time period, the node tears down the connection. Range: 10–300. Default = 300.

Related Commands

dsppnni-svcc-rcc-timer

Attributes

Log: log	State: active	Privilege: SUPER_GP

This command is not supported by the current release.

cnfpnni-timer

Configure PNNI Timers

The **cnfpnni-timer** command specifies the initial PNNI timer values and significant change thresholds of a PNNI node in the network.

Note Use of this command must be carefully planned because modifying a timer can significantly change network efficiency.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnni-timer <node-index>
[-ptseHolddown ptse-holddown]
[-helloHolddown hello-holddown]
[-helloInterval hello-interval]
[-helloInactivity-factor hello-inactivity-factor]
[-horizontalLinkInactivityTime horizontal-link-inactivity-time]
[-ptseRefreshInterval ptse-refresh-interval]
[-ptseLifetimeFactor ptse-lifetime-factor]
[-retransmitInterval retransmit-interval]
[-ptseDelayedAckInterval ptse-delayed-ack-interval]
[-avcrPm avcr-pm]
[-avcrMt avcr-mt]
[-cdvPm cdv-pm]
[-ctdPm ctd-pm]
```

Syntax Description

node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default = 1
-ptseHolddown	The holddown value is the time the switch waits before it broadcasts PTSEs. The increments are tenths of a second. For example, **-ptseHolddown** 1 means 0.1 second, and **-ptseHolddown** 10 means 1 second, and so on. Range: 0.1–1000 Default: 10 (1 second)

-helloHolddown	Specify the initial value for the Hello hold down timer that the node uses to limit the rate at which it sends Hellos. Units: 100 ms. 1 = 0.1 seconds Range: 0.1–1000 Default = 10, (1 second)
-helloInterval	Specify the initial value for the Hello timer in seconds. The value limits the rate at which the node sends Hellos. Range: 1–300 seconds. Default = 15 seconds.
-helloInactivityFactor	The product of the *hello-inactivity-factor* and the peer neighbor *hello-interval* is the maximum time (in sec) that the neighbor is considered alive after the last reception of a Hello packet. Range: 1–50 seconds. Default = 5 seconds.
-ptseRefreshInterval	Specify the initial time allowed for the PTSE to re-originate specified in seconds. Range: 30–1800 seconds. Default = 1800 seconds.
-ptseLifetimeFactor	Specify the value for the lifetime multiplier, expressed as a percentage. The product of it and the *ptse-refresh-interval* is the initial value of the remaining lifetime of a self-created PTSE. Range: 101–1000 per cent. Default = 200.
-retransmitInterval	Specify the period between retransmissions of unacknowledged DS, PTSE request, and PTSP specified in seconds. Range: 5–60 seconds. Default = 5 seconds.
-ptseDelayedAckInterval	Specify the minimum interval between transmissions of delayed PTSE acknowledgment packets. Units: 100 ms. 1 = 0.1 seconds. Range: 1–10 seconds. Default = 10, i.e., 1 second.
-avcrPm	Specify the proportional multiplier used in the algorithms that determine significant change for available cell rate (AvCR) parameters. *avcr-pm* is expressed as a percentage. Range: 1–99 per cent. Default = 50 per cent.

-avcrMt	Specify the minimum threshold used in the algorithms that determine significant change for available cell rate (AvCR) parameters which are expressed as a percentage. You can change this value to minimize the overhead created by advertisements triggered by AVCR changes. *avcr-mt* is expressed as a percentage. Range: 1–99 per cent. Default = 3 per cent.
-cdvPm	Specify the proportional multiplier used in the algorithms that determine significant change for Cell Delay Variation (CDV) parameters which are expressed as a percentage. is the variation of delay between cells, measured peak to peak.You can change this value to minimize the overhead created by advertisements triggered by CDV changes. *cdv-pm* is expressed as a percentage. Range: 1–99 per cent. Default = 25 per cent.
-ctdPm	Specify the proportional multiplier used in the algorithms that determine significant change for cell transfer delay (CTD) parameters which are expressed as a percentage. You can change this value to minimize the overhead created by advertisements triggered by changes to CTD values. *ctd-pm* is expressed as a percentage. Range: 1–99 per cent. Default = 50 per cent.

Usage Guidelines

PTSE packets update the peer group when network changes occur. Your network should run fine with the defaults PTSE timing parameters. Networks that have properties significantly different from the norm may perform better if you optimize some of the PTSE parameters, but you should change these parameters carefully, and test the network before introducing live traffic.

For example, if no one is permitted to change a network topology (perhaps for test purposes), you may decide that the default **-ptseHolddown** value (1 sec) is too small, causing sequential broadcasts of identical PTSE packets. Further reason for making such a change would be given if the network was a small and it's connections were short. Inversely, it may warrant a smaller **-ptseHolddown** value if many changes are being made to a network, or if it was large, or if it had many long connections. These are only examples—these changes should only be made be network experts.

Related Commands

dsppnni-timer

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

This example shows the **cnfpnni-timer** command line that is set with the following parameter values:

- The node broadcasts Hello packets every 1.2 seconds.
- The node broadcasts Hello packets every 1.2 seconds.
- The node Hello timer is 15 seconds.
- The node *hello-inactivity-factor* is 5 seconds.
- The node recreates PTSEs every 1800 seconds (30 minutes).
- The node evaluates the initial PTSE lifetime by multiplying the ptse-refresh-interval (1800 seconds) by 200.
- The node retransmits every 5 seconds.
- If the node broadcasts a PTSE and the PTSE is not acknowledged, the node waits 1 second to rebroadcast its PTSE.
- The node multiplies an algorithm by 3% to determine the value that qualifies as a significant change for AvCR parameters.
- The node multiplies an algorithm by 50% to determine the value that qualifies as the minimum threshold for significant change of AvCR parameters.
- The node multiplies an algorithm by 25% to determine the value that qualifies as a significant change for CDV parameters.
- The node multiplies an algorithm by 50% to determine the value that qualifies as a significant change for CTD parameters.

You can confirm your settings with the related command **dsppnni-timer**.

```
SanJose.7.PXM.a > cnfpnni-timer 1 -ptseHolddown 120 -helloHolddown 120 -helloInterval 15
-helloInactivity-factor 5 -horizontalLinkInactivityTime 10 -ptseRefreshInterval 1800
-ptseLifetimeFactor 200 -retransmitInterval 50 -ptseDelayedAckInterval 10 -avcrPm 3
-avcrMt 50 -cdvPm 25 -ctdPm 50

SanJose.7.PXM.a > dsppnni-timer 1
node index: 1
Hello holddown(100ms)... 12 PTSE holddown(100ms)... 12
Hello int(sec).......... 15 PTSE refresh int(sec).. 1800
Hello inactivity factor. 5 PTSE lifetime factor... 200
Retransmit int(sec)..... 5
AvCR proportional PM.... 3 CDV PM multiplier...... 25
AvCR minimum threshold.. 50 CTD PM multiplier...... 50
Peer delayed ack int(100ms).................. 10
Logical horizontal link inactivity time(sec).. 10

Geneva.7.PXM.a >
```

dbgpnni

Debug PNNI Messages—specify which PNNI debug messages are enabled.

The **dbgpnni** command lets you specify the types of debug messages that appear in the display of the **dsppnni-dbg** command.

Note The selections you make with this command can significantly affect network performance. The simultaneous dumping of multiple types of debug messages can increase the overhead more than incrementally as each dump is added. Consider the options individually rather than simultaneously.

Cards on Which This Command Runs

PXM45

Syntax

```
dbgpnni
[-all {on | off}]
[-hello {on | off}]
[-election {on | off}]
[-nbr {on | off}]
[-itf {on | off}]
[-timer {on | off}]
[-lgn {on | off}]
[-spt {on | off}]
[-nodereachability {on | off}]
[-address {on | off}]
[-itdb {on | off}]
[-ra {on | off}]
[-cp {on | off}]
[-linkselection {on | off}]
```

Syntax Description

-all	Specify whether all types of debug messages go to the console. on: All types of debug messages go to the console. off: Only the debug message-types specified on by the other **dbgpnni** parameters are active. Default: none
-timer	Specify whether the timer debug messages go to the console. on: The timer debug messages go to the console. off: The timer debug messages do not go to the console. Default: none
-hello	Specify whether the hello debug messages go to the console. on: The hello debug messages go to the console. off: The hello debug messages do not go to the console. Default: none

-nbr	Specify whether the PNNI neighbor FSM debug messages go to the console. on: The PNNI neighbor FSM messages go to the console. off: The PNNI neighbor FSM debug messages do not go to the console. Default: none
-election	Specify whether the PNNI PGL election debug messages go to the console. on: The PNNI PGL election messages go to the console. off: The PNNI PGL election debug messages do not go to the console. Default: none
-nodereachability	Specify whether the debug messages about PNNI node reachability computation go to the console. on: The PNNI node reachability computation messages go to the console. off: The PNNI node reachability computation is not active. Default: none
-itf	Specify whether the interface FSM debug messages go to the console. on: The interface FSM messages go to the console. off: The interface FSM debug messages do not go to the console. Default: none
-address	Specify whether the debug messages for address handling go to the console. on: The debug handling on addresses is active. off: The debug handling on addresses is not active. Default: none
-lgn	Specify whether the PNNI logical group node debug messages go to the console. on: The PNNI logical group node messages go to the console. off: The PNNI logical group node debug messages do not go to the console. Default: none
-itdb	Specify whether the debug messages about the internal topology database (IDB) go to the console. on: The debug messages about the IDB go to the console. off: The debug messages about the IDB do not go to the console. Default: none
-cp	Specify whether the control point debug messages go to the console. on: The control point debug messages go to the console. off: The control point debug messages do not go to the console. Default: none

-spt	Specify whether the shortest path tree debug messages go to the console. on: The shortest path tree debug messages go to the console. off: The shortest path tree debug messages do not go to the console. Default: none
-ra	Specify whether the route agent debug messages go to the console. on: The route agent debug messages go to the console. off: The route agent debug messages do not go to the console. Default: none
-linkselection	Specify whether the link selection debug messages go to the console. on: The link selection debug messages go to the console. off: The link selection debug messages do not go to the console. Default: none

Related Commands

dsppnni-dbg

Attributes

Log: log	State: active	Privilege: CISCO_GP

Example

Specify the following PNNI debug message-options:

- Interface debugging is enabled.
- Address handling debugging is enabled.

Use **dsppnni-dbg** to check which debug messages types appear on the terminal.

```
SanJose.7.PXM.a > dsppnni-dbg
pnni debugging option:

hello   election   nbr   itf   timer   lgn   spt   node reachability
-----   --------   ---   ---   -----   ---   ---   -----------------
off     off        off   off   off     off   off   off

address   itdb   ra    cp    link selection
-------   ----   ---   ---   --------------
off       off    off   off   off


SanJose.7.PXM.a > dbgpnni -itf on -address on


SanJose.7.PXM.a > dsppnni-dbg
pnni debugging option:

hello   election   nbr   itf   timer   lgn   spt   node reachability
-----   --------   ---   ---   -----   ---   ---   -----------------
off     off        off   on    off     off   off   off

address   itdb   ra    cp    link selection
-------   ----   ---   ---   --------------
on        off    off   off   off
```

In the second example, the parameters are changed as follows:

- Interface debugging is disabled.
- Address handling debugging is disabled.

```
SanJose.7.PXM.a > dbgpnni -itf off -address off


SanJose.7.PXM.a > dsppnni-dbg
pnni debugging option:

hello   election   nbr   itf   timer   lgn   spt   node reachability
-----   --------   ---   ---   -----   ---   ---   -----------------
off     off        off   off    off     off   off   off

address   itdb   ra    cp    link selection
-------   ----   ---   ---   --------------
off       off    off   off   off


SanJose.7.PXM.a >
```

delpnni-node

Delete PNNI Node–delete a PNNI node from the PNNI network topology.

The **delpnni-node** command removes a PNNI node from the PNNI network topology. A node that this command deletes can be restored with the related command **addpnni-node**. Confirm the deletion of nodes with the **dsppnni-node** command.

Cards on Which This Command Runs

PXM45

Syntax

delpnni-node <*node-index*>

Syntax Description

node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default: 1

Related Commands

addpnni-node, **cnfpnni-node, delpnni-node**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Delete the node with the *node-index* of 3. Execute **dsppnni-node** and specify a node index of 3. The last line of the **d** display shows the error "node does not exist," indicating you successfully deleted the node.

```
SanJose.7.PXM.a > delpnni-node 3
SanJose.7.PXM.a > dsppnni-node 3

   Unknown Error Code
Syntax: dsppnni-node [node-index]

        nodeIndex -- node-index: 32 bit number starting from 1, Optional parameter

        possible errors are:
node does not exist

SanJose.7.PXM.a >
```

delpnni-summary-addr

Delete PNNI Summary Address—delete a PNNI summary address from the node.

The **delpnni-summary-addr** command deletes a PNNI summary address for a PNNI node.

Cards on Which This Command Runs

PXM45

Syntax

delpnni-summary-addr *<node-index> <address-prefix> <prefix-length>*

Syntax Description

node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default: 1
address-prefix	The PNNI summary address. Default: none
prefix-length	The length of the *address-prefix* in bits is less than or equal to 152. In the current release, the zero-length PNNI summary address is not supported. Default: none

Related Commands

addpnni-summary-addr, **dsppnni-summary-addr**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Delete a summary address, as follows:

- The node index is 1.
- The node address prefix is 47.0091.8100.0000.0030.9409.f1f1.
- The length of the address prefix is 104 bits.

If necessary, use **dsppnni-summary-addr** to confirm the deletion.

```
SanJose.7.PXM.a > delpnni-summary-addr 1 47.0091.8100.0000.0030.9409.f1f1 104
SanJose.7.PXM.a >
```

dsppnni-bn-path

Display PNNI Border Node Paths—display the PNNI border node paths.

This debugging command displays the border node-to-border node paths of the immediate child-peer-group of the logical group nodes (LGN).

Note The command applies to multi-peer groups only.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-bn-path` *`<node-index>`*

Syntax Description

node-index	The node index indicates the relative level of the logical node within the hierarchy on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default: none

Contents of the dsppnni-bn-path Output

node index	The range is 1–10.
source node IDB index	The node identifier within the internal data base (IDB). Range: 1–2147483648
source node ID	The node ID of the source.
destination node IDB index	The node ID of each destination node.

destination node ID	The node ID of each destination.
metrics for the path	For each class of service (CBR, rt-VBR, nrt-VBR, ABR, UBR), the configuration for each of the following routing metrics: • Administrative weight (AW) • Maximum cell rate (MaxCR) • Available cell rate (AvCR) • Cell transfer delay (CTD) • Cell delay variation (CDV) • Cell loss priority, first leaky bucket (CLP0) • Cell loss priority, second leaky bucket (CLP0+1)

Related Commands

None

Attributes

Log: no	State: active, standby	Privilege: ANYUSER

Example

Enter **dsppnni-db-path** and specify node index 2.

```
mpg2.7.PXM.a > dsppnni-bn-path 2

node index:2

source node IDB index   source node id
---------------------   --------------------------------------------------
                    1   56:160:47.00918100000000003071f80e56.003071f80e56.01

destination node IDB index   destination node id
--------------------------   --------------------------------------------------
                        12   56:160:47.00918100000000003071f80e52.003071f80e52.01

            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MaxCR    250000   250000   250000   250000   250000
AvCR     248759   248759   248759   248759   248759
CTD          17       17       17       17       17
CDV        4167    52954    52954   104912   104912
CLR0          8        8        8        8        8
CLR0+1        8        8        8        8        8
```

dsppnni-bypass

Display PNNI Bypasses—display the PNNI complex node bypass table.

This debugging command displays the PNNI bypass table for a logical group node (LGN) that uses the complex node representation of its peer group. The bypass table contains the topology and Hello information of every node in the peer group.

Note This command applies to multi-peer groups only.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-bypass <node-index>`

Syntax Description

node-index	The node index points to the LGN of this peer group. It indicates the relative level of the node within the hierarchy on the switch. The range is 1–10, and the lowest level is 1. Range: 1–10 Default: 1

Display Contents

The display contains the following fields for each node.

node-index	The node index indicates the relative level of the logical node within a multi-peer group on the switch. The supported range is 1–10, and the lowest level is 1
Input port Id	The logical form of port identifier at the input. Range: 1–2147483648.
Output port Id	The logical form of port identifier at the output. Range: 1–2147483648.
ptse id	Display the unique identifier assigned to the PNNI topology state elements PTSE. *ptse-id* is the 32 bit numeric node identifier assigned by the software—it is not user-configurable.
flags	Although a hexadecimal flag appears, it applies to software debugging by Cisco engineers and so is not described here.

Related Commands

None

Attributes

Log: nolog State: active, standby Privilege: ANYUSER

Example

Display the bypass table for an LGN using complex node representation in the peer group.

```
Geneva.7.PXM.a > dsppnni-bypass
node index: 1
Input portId........ 376 Output portId........ 399
Ptse id............. 12 Flags................ a3

          CBR  RTVBR NRTVBR    ABR    UBR
-----  ------ ------ ------ ------ ------
AW       5040   5040   5040   5040   5040
MCR         0      0      0      0      0
AvCR   100000 100000 100000 100000 100000
CTD         0      0      0      0      0
CDV         0      0      0      0      0
CLR0        0      0      0      0      0
CLR0+1      0      0      0      0      0
CRM        10     10     10     10     10
VF          5      5      5      5      5

Geneva.7.PXM.a >
```

dsppnni-dbg

Display PNNI Debug Flags—display the settings of all PNNI debug flags.

The **dsppnni-dbg** command displays which PNNI debug options are enabled and which are disabled.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-dbg`

Display Contents

This section describes the content of the display for each node. The right column shows the label for each value that appears in the **dsppnni-dbg** command. The left column maps each value to the corresponding keyword in the **dbgpnni** command, and explains the argument function.

Hello	Display the flag that indicates whether the Hello packet debug is enabled. on: the Hello packet debug is enabled. off: the Hello packet debug is disabled.
election	Display the flag that indicates whether the peer group election debug is enabled. on: the peer group election debug is enabled. off: the peer group election debug is disabled.
nbr	Display the flag that indicates whether the neighbor debug is enabled. on: the neighbor debug is enabled. off: the neighbor debug is disabled.
itf	Display the flag that indicates whether the interface debug is enabled. on: the interface debug is enabled. off: the interface debug is disabled.
timer	Display the flag that indicates whether the timer debug is enabled. on: the timer debug is enabled. off: the timer debug is disabled.
lgn	Display the flag that indicates whether the logical node (LGN) debug is enabled. on: the LGN debug is enabled. off: the LGN debug is disabled.
spt	Display the flag that indicates whether the logical node SPT debug is enabled. on: the SPT debug is enabled. off: the SPT debug is disabled.

node reachability	Display the flag that indicates whether the node reachability debug is enabled. on: the node reachability debug is enabled. off: the node reachability debug is disabled.
address	Display the flag that indicates whether the addressing debug is enabled. on: the addressing debug is enabled. off: the addressing debug is disabled.
itdb	Display the flag that indicates whether the internal data base debug is enabled. on: the internal data base debug is enabled. off: the internal data base debug is disabled.
ra	Display the flag that indicates whether the route agent debug is enabled. on: the route agent debug is enabled. off: the route agent debug is disabled.
cp	Display the flag that indicates whether the CP debug is enabled. on: the CP debug is enabled. off: the CP debug is disabled.
link selection	Display the flag that indicates whether the link selection debug is enabled. on: the link selection debug is enabled. off: the link selection debug is disabled.

Related Commands

dbgpnni

Attributes

Log: nolog	State: active	Privilege: CISCO_GP

Example

Display the active PNNI debug options.

```
Unknown.1.1.PXM45.a > dsppnni-dbg

pnni debugging option:

hello election nbr itf timer lgn spt   node reachability
----- -------- --- --- ----- --- ---   -----------------
off   off      off off off   off off   off

address   itdb    ra    cp   link selection
-------   ----    ---   ---  --------------
off       off     off   off  off

Geneva.7.PXM.a >
```

dsppnni-election

Display PNNI Election—display information for election a peer group leader.

The **dsppnni-election** command displays parameters and current status related to the election of a peer group leader (PGL). This command and related information applies to multi-peer groups only. The **cnfpnni-election** command lets you specify election parameters.

In a multi-peer group environment, each peer group can elect one PGL. Such an election takes place for every level of the hierarchy. (For example, if three levels exist, three PGL elections occur.)

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-election
[node-index]
```

Syntax Description

node-index	The system-generated node index indicates the *relative* level of the logical node within a multi-peer group on the switch. The range is 1–10. The lowest node is 1, and the highest is 10. (Note that *node index* is inversely related to the *node level*, which you specify through such commands as **cnfpnni-node**, for example, and has a range of 1–104. The lowest node level is 104. When you add a logical node to the hierarchy on a switch, the system generates the node index.) Range: 1–10 Default: 1

Objects Displayed

The following parameters are displayed for each node.

Node-index	The node index has a range of 1–10.
PGL state	OperNotPGL, OprPGL, starting, and so on.??
Priority	This node’s leadership priority in a peer group.
Initialization time	The number of seconds that this node delays advertising its choice of preferred PGL.

Override delay	The number of seconds that this node waits for itself to be declared the preferred PGL by unanimous agreement among its peers.
Re-election time	After losing connectivity to the current peer group leader, the number of seconds that this node waits before re-starting the process of electing a new peer group leader.
Preferred PGL	The ID of the node that should be the PGL according to the current node. This choice weighs information on leadership priorities and node IDs that it receives from the PTSEs.
PGL	The node in the peer group that has been elected PGL
Active parent node ID	The node ID of the LGN.

Related Commands

cnfpnni-election, **cnfpnni-node**, **dsppnni-node**, **dsppnni-node-list**, **dsppnni-summary-addr**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the information about peer group leader election for all nodes in the hierarchy.

```
8850_NY.7.PXM.a > dsppnni-election

node index: 1
   PGL state......     OperNotPgl     Init time(sec).......        15
   Priority.......            150     Override delay(sec)..        30
                                      Re-election time(sec)        15
   Pref PGL...............56:160:47.00918100000020004c113ba75.0004c113ba75.01
   PGL....................56:160:47.00918100000020004c113ba75.0004c113ba75.01
   Active parent node id..0:0:00.000000000000000000000000.000000000000.00



node index: 2
   PGL state......       Starting     Init time(sec).......        15
   Priority.......              0     Override delay(sec)..        30
                                      Re-election time(sec)        15
   Pref PGL...............0:0:00.000000000000000000000000.000000000000.00
   PGL....................0:0:00.000000000000000000000000.000000000000.00
Active parent node id..0:0:00.000000000000000000000000.000000000000.00
```

dsppnni-idb

Display PNNI Internal Data Base—display the PNNI link-state information for the node.

The **dsppnni-idb** command applies solely to debugging. It lets you see all the nodes and links that the current node has discovered. If a node or link should be in the internal database (IDB) but is missing, you can check the PTSEs (**dsppnni-ptse**) to begin tracing the missing topology information.

An IDB stores all the logical nodes known to the local node (its own levels and the network nodes in each logical node's view) and the outgoing links from all of them. The IDBs are the source of all address and routing tables in the peer group. When a node advertises PNNI topology state elements (PTSEs), the updated information goes into the IDB updates. The system address table, local network reachable address table, background routing tables, and PNNI summary address table receive updates from the IDB as appropriate.

The **dsppnni-idb** command can display all the contents or a subset of the IDB. You can specify the granularity of the display by using the optional parameters:

- If you enter **dsppnni-idb** with no parameters, the display shows the internal topology database of all nodes in the peer group.
- If you specify a node index, the display shows the internal topology database of all nodes that are visible to the local, logical node with the specified index.
- If you specify a node number, the display shows the internal topology database for a specific, remote node within the peer group. To see the valid node numbers for nodes in the peer group, first use **dsppnni-node-list**.
- If you specify a port ID after specifying a node number, the display shows the internal topology database of that specific port on that remote node.

Cards on Which This Command Runs

PXM45

Syntax

dsppnni-idb [*node-index*] [-**nodeNumber** *node-number* [-**portId** *port-id*]]

Syntax Description

node-index	The node index is the system-generated number of the *local* logical node. In a multi-peer group hierarchy, the range is 1–10. In the current release, the only node index is 1. Range: 1–10
-**nodeNumber**	A number that uniquely identifies a node in the network. For a list of the remote node numbers that are visible to the local node, use **dsppnni-node-list**. Range: 1–256
-**portId**	The logical number for a PNNI port. Use this optional parameter if you have specified a node number (-**nodeNumber**) and want to narrow the scope of the display. You can obtain the logical number for the port ID by executing **dsppnports** for all PNNI ports or **dsppnport** *a:b.c.d* for a specific port (where a,b,c, and d are the values corresponding to the physical portID. For more details, see the "PNNI Format" section on page 5-5.) Range: 1–2147483648

Related Commands

none

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the link-state information for the following:

- Node index: 1
- Node number: 1
- Logical port ID: 16848901

```
Geneva.7.PXM.a > dsppnni-idb 1 -nodeNumber 1 -portId 16848901
node index:1
   Local port id.......  16848901     Remote port id.......  16848901
   Local link index....         1     Remote link index....         1
   Local node number...         1     Remote node number...         2
   PGL node index......         0     LGN node index.......         0
   Transit restricted..       off     Complex node.........       off
   Branching restricted        on     PGL..................     false
   Ancestor............     false     Border node..........     false
   VP capable..........      true     Link type...........horizontal
   Non-transit for PGL election..       off
   node id..............56:160:47.00918100000000107be92f46.00107be92f46.01
   node name.............pswmgx2-2

Geneva.7.PXM.a >
```

dsppnni-inducing-uplink

Display PNNI Inducing Uplink—display the PNNI inducing uplink database.

The **dsppnni-inducing-uplink** command displays the uplink-inducing database. The only application of **dsppnni-inducing-uplink** is debugging.

Note This command applies to multi-peer groups only, so it has no purpose in the current release.

The display shows:

- Child node index number
- Token (if configured)
- Uplink node ID—the ID of the node
- Uplink ATM address
- Uplink peer group ID
- Routing metrics of the uplink

The child node number is the number of a node at a lower level (as shown by **dsppnni-node-list**) from which the uplink comes. The child port ID is the local port ID of the child node from which the uplink comes. You can see the details of this uplink by executing **dsppnni-idb**. (In the display for **dsppnni-idb**, the child node index is the local node number.)

The uplink node or *upnode* is the node at the upper end of the uplink. It is the neighboring peer of the ancestor of the node from which the uplink originates.

The extent of the **dsppnni-inducing-uplink** display depends on whether you specify an individual logical node in the hierarchy, as follows:

- If you specify a node index, the display shows the PNNI-inducing uplink database of a specific logical node on the switch.
- If you do not specify a node index, the command displays the PNNI-inducing uplink database for each logical node on the switch.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-inducing-uplink [`*`node-index`*`]`

Syntax Description

node-index	The *node-index* specifies the logical node on the switch. Range: 1–10 Default: 1

Related Commands

None

Attributes

Log: nolog State: active, standby Privilege: ANYUSER

Example

Display the inducing uplink for the level whose node index is 2.

```
mpgses3.2.PXM.a > dsppnni-inducing-uplink 2

node index:2
   Token................         0      Child node portId....     66560
   Child node number....         1
   Upnode id.............32:56:47.00918100110000000000001.003071f80e56.00
   Upnode ATM addr.......47.009181000000003071f80e56.003071f80e56.02
   Upnode PG id..........32:47.00.9181.0000.0000.0000.0000.00

               CBR      RTVBR    NRTVBR   ABR      UBR
               ------   ------   ------   ------   ------
      AW        10000    10000    10000    10000    10000
      MaxCR    348207   348207   348207   348207   348207
      AvCR     347419   347419   347419   347419   347419
      CTD          17       17       17       17       17
      CDV        4167    52954    52954   104912   104912
      CLR0          8        8        8        8        8
      CLR0+1        8        8        8        8        8
```

dsppnni-intf

Display PNNI Interface—display the PNNI aggregation token and AW for a port.

This command displays the following information for a PNNI port:

- Aggregation token. The current release does not support the aggregation token.
- Administrative weight (AW).
- The logical port identifier assigned to the physical port identifier.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-intf` <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 5-5. Default: none

Display Contents

The following parameters are displayed for each node. The right column shows the label for each value that appears in the **dsppnni-intf** command. The left column maps each value to the corresponding keyword in the **cnfpnni-intf** command and explains the argument function.

Physical port Id	Display the physical PNNI identifier on the interface. The *portid* variable is entered and displayed as a numeric string formatted as [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. For more details including the ranges, see PNNI Format.
Logical port Id	Display the logical PNNI port identifier. The **-portId** parameter displays the logical PNNI port identifier on the interface. Range: 1–2147483648.
Aggr token	Display the 32 bit number used for link aggregation purpose.
AW-NRTVBR	Display the AW for non-real-time variable bit rate (nrtvbr) connections on this interface. nrtvbr accounts for the bursty traffic that is caused by some non-real-time applications. This category is characterized in terms of a PCR, SCR, and MBS. Range: 0–4194304

AW-CBR	Display the AW for constant bit rate (CBR) connections on this interface. While a cbr connection is active, this option limits its bit rate to a static value that remains available until the connection is torn down. The bit rate is characterized by the peak cell rate (PCR) value. Range: 0–4194304
AW-ABR	Display the AW for available bit rate (ABR) connections on this interface. While an ABR connection is active, that connection changes its permitted bit rate (bandwidth) in response to ATM layer transfer characteristics. PNNI periodically polls for the unused bandwidth at the ATM level, then adjusts the ABR connection bit rate in direct relation to the available bandwidth. Displays the 24 bit number AW for ABR on this interface. Range: 0–4194304
AW-RTVBR	Display the AW for real-time variable bit rate (rt-vbr) connections on this interface. rtvbr is intended for real-time applications requiring tightly constrained delay and delay variation (voice/video applications). rtvbr is characterized by peak cell rate (PCR), sustainable cell rate (SCR), and maximum burst size (MBS). Range: 0–4194304
AW-UBR	Display the AW (QoS) used for unspecified bit rate (UBR) connections—this includes switched virtual connection (SVC) ping connections. While a UBR connection is active, this option limits only the maximum bit-rate (bandwidth) of the connection—no bit rate is guaranteed. Range: 0–4194304

Display Contents

PNNI includes a topology state routing protocol, which advertises detailed information about the peer groups links and nodes. Links and nodes are assigned metrics and attributes that can be used to diagnose or tune network behavior.

The administrative weight (AW) is the cost to traffic that traverses a port. The AW for a path is the sum, in both directions, of the individual AWs the egress of each port on the path.

The AW can be specified on the interface and by the service class (or QoS class), and it is associated with each port. AW is a defining factor when routes are selected. The AW parameters influence how PNNI selects paths in the peer group and therefore how it distributes each SVC and SPVC. PNNI route selection can also key on AW to exclude certain links from routing. The application of such exclusion can be to defining a backup link for use only when no bandwidth is available on the primary link.

Related Commands

cnfpnni-intf

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display the interface configuration for port 4:1.1:11. The current release does not support the link aggregation parameter (displayed as “Aggr token”), so it appears as a 0.

```
SanJose.7.PXM.a > dsppnni-intf 4:1.1:11

Physical port id: 4: 1.1:11        Logical port id:   17045515
   Aggr token..........         0     AW-NRTVBR...........      5040
   AW-CBR..............      5040     AW-ABR..............      5040
   AW-RTVBR............      5040     AW-UBR..............      5040

SanJose.7.PXM.a >
```

dsppnni-link

Display PNNI Link Table–display the values of the PNNI link table.

The **dsppnni-link** command displays the parameters of all PNNI links.

- If you specify a node index and a port ID, the command displays information about that specific PNNI link.
- If you specify only a node index, the display shows all PNNI links attached to that node.
- If you specify nothing, the command displays all links attached to all PNNI nodes in the network.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-link
[node-index [port-id]]
```

Syntax Description

node-index	A system-generated value that corresponds to a logical node in an MPG hierarchy. For every PNNI node that you add through CWM or the **addpnni-node** command, the system associates the next available integer in the range 1–10. **Note** In the current release, the value of the *node-index* is always 1. Range: 1–10
portID	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 5-5.

Display Contents

The **dsppnni-link** command displays the address, link, and Hello packet information of each link. In a multi-peer group, it also displays the upnode ATM address and node ID. For an explanation of upnode, see the description of **dsppnni-inducing-uplink**.

Related Commands

dsppnni-link-selection

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Specify node index 1 and port 1:1.2:2.

```
p2spvc5.7.PXM.a > dsppnni-link 1 1:1.2:2

node index   :1
Local port id:  16848898          Remote port id:  16848898
Local Phy Port Id:1:1.2:2
   Type. lowestLevelHorizontalLink     Hello state....... twoWayInside
   Derive agg...........         0     Intf index...........  16848898
   SVC RCC index........         0     Hello pkt RX.........         2
                                       Hello pkt TX.........         2
   Remote node name.......p2spvc6
   Remote node
id.........56:160:47.00918100000000309409f1ef.00309409f1ef.01
   Upnode
id..............0:0:00.000000000000000000000000.000000000000.00
   Upnode ATM addr........00.000000000000000000000000.000000000000.00
   Common peer group id...00:00.00.0000.0000.0000.0000.0000.00
```

dsppnni-link-selection

Display PNNI Link Selection—display the PNNI link setting, physical port identifier, and logical port identifier.

The **dsppnni-link-selection** command displays the link selection and both the physical and logical identifiers for that link. Refer to the description of **cnfpnni-link-selection** for information about the criteria PNNI uses to choose between two parallel links.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-link-selection` <*portid*>

Syntax Description

portid	The PNNI physical port identifier. *portid* is entered and displayed as a 32-bit string formatted as [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. For more details, see PNNI Format.

Display Contents

The display shows the following information for each node.

physical port id	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 5-5.
link selection	The ASCII string displaying the link routing policy.
logical port id	The PNNI port identifier in the form of a logical number. Range: 1–2147483648

Related Commands

dsppnni-link

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Displays the link selection for a parallel link on port 4:1.1:11. This link uses the default of minaw. Note that the display also shows the logical port number for the physical port ID.

```
SanJose.7.PXM.a > dsppnni-link-selection 4:1.1:11

physical port id:        4:1.1:11     link selection: minaw
 logical port id:        17045515



SanJose.7.PXM.a >
```

dsppnni-mtu

Display PNNI Maximum Transfer Unit—display maximum supported size of the PNNI data packet.

The **dsppnni-mtu** command displays the maximum PNNI packet size in number of bytes. This command is primarily for configuring internetwork compatibility, but you can use it in lab trials to test the affect of various packet sizes on the performance of the peer group.

Use **cnfpnni-mtu** to specify the PNNI packet size configuration.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-mtu
```

Display Contents

The following parameters are displayed for each node. The right column shows the label for each value that appears in the **dsppnni-mtu** command. The left column maps each value to the corresponding keyword in the **cnfpnni-mtu** command, and explains the argument function.

max packet size	The value of the max transmit unit *mtu* in number of bytes. Range: 1024–8192

Related Commands

dsppnni-idb, **cnfpnni-mtu**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Displays the maximum packet size.

```
SanJose.7.PXM.a > cnfpnni-mtu 3002

SanJose.7.PXM.a > dsppnni-mtu

max packet size : 3002


SanJose.7.PXM.a >
```

dsppnni-neighbor

Display PNNI Neighbor—display all PNNI nodes that directly connect to this node.

The **dsppnni-neighbor** command displays all the PNNI nodes that are directly connected to the switch.

- If you specify: both *node-index* and *rmt-node-id*, the command displays information about the *rmt-node-id* neighbors.
- If you specify nothing, the command displays all neighbors attached this switch.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-neighbor
[node-index [rmt-node-id]]
```

Syntax Description

Note that the optional parameters are nested.

node-index	The system-generated node index specifies the relative level of the PNNI logical node within the hierarchy on the switch. the node identifier. In the current release, the only node index is 1. Range: 1–10
rmt-node-id	The node index for a remote node is the index assigned to a neighboring node.

Display Contents

The following parameters are displayed for each node.

node-index	The system-generated node index in the range 1–10. In the current release, the only node index is 1.
node name	The name of the neighboring node (assigned through the **cnfname** command).
Remote node id	The PNNI logical node identifier (node ID). The *node-id* consists of the following logical elements, starting at the most significant byte: • The level of the PNNI node within the hierarchy. (See the description of the *level* parameter.) • The number of bits in the ATM address. The number is 160 for an NSAP address because the ATM address of the node is always 20 bytes. For an E.164 address, this field is decimal 15. • The ATM address portion of the peer group ID (20 8-bit, hexadecimal bytes).

Neighbor state	FULL??
Port count	The number of ports.
SVC RCC index	The index for the SVC routing control channel.
RX DS pkts	The number of received signal packets in the receive direction. The current release does not support SVCC RCC, so this value is '0.'
TX DS pkts	The number of transmitted signal packets in the transmit direction.
RX PTSP pkts	The number of PNNI topology state packets in the receive direction.
TX PTSP pkts	The number of PNNI topology state packets in the transmit direction.
RX PTSE req pkts	The number of PNNI topology state element (PTSE) request packets in the receive direction.
TX PTSE req pkts	The number of transmitted PTSE request packets in the transmit direction.
RX PTSE ack pkts	The number of received PTSE acknowledgment packets in the receive direction.
TX PTSE ack pkts	The number of transmitted PTSE acknowledgment packets in the transmit direction.

Related Commands

None

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display information about all neighboring PNNI nodes (with no optional parameters).

```
Geneva.7.PXM.a > dsppnni-neighbor

node index    : 1

node name     : Paris
Remote node id: 56:160:47.00918100000000107b65f27c.00107b65f27c.01
Neighbor state: FULL
   Port count..........         4     SVC RCC index.......         0
   RX DS pkts..........         3     TX DS pkts..........         3
   RX PTSP pkts........      6032     TX PTSP pkts........      2061
   RX PTSE req pkts....         2     TX PTSE req pkts....         1
   RX PTSE ack pkts....       345     TX PTSE ack pkts....      2282

node index    : 2

node name     : SanJose
Remote node id: 56:160:47.00918100000000309409f1f1.00309409f1f1.01
Neighbor state: FULL
   Port count..........         2     SVC RCC index.......         0
   RX DS pkts..........         4     TX DS pkts..........         3
   RX PTSP pkts........     23107     TX PTSP pkts........     32978
   RX PTSE req pkts....         3     TX PTSE req pkts....         0
   RX PTSE ack pkts....     13673     TX PTSE ack pkts....     12532

Geneva.7.PXM.a >
```

dsppnni-node

Display PNNI Node—display the PNNI node information address and routing information.

The **dsppnni-node** command displays the PNNI logical node information on the local switch. If you do not provide an index number, the output shows all logical nodes on the switch. However, in the current release, the display always shows only one logical node—with index number 1.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-node
[node-index]
```

Syntax Description

node-index	The *node-index* identifies a logical node in relation to other nodes in the hierarchy. This node index applies locally within the switch.In the current release, the only local node index is the default of 1, so you do not actually have to enter it. Range: 1–10 Default: 1

Related Commands

addpnni-node, **cnfpnni-node**

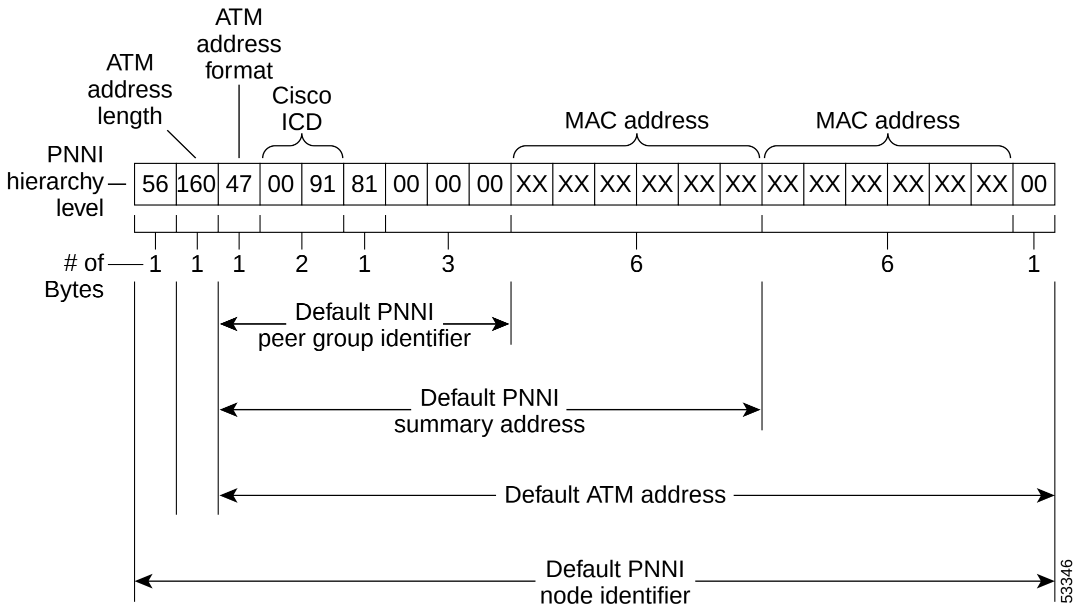
Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Display Contents

Cisco factory-set defaults for address prefixes and the peer group ID share field-values with the ATM address. (See Figure 5-7.)

Figure 5-7 Cisco Factory-shipped Defaults for PNNI Peer Group Identifier, PNNI Summary Address, ATM Address, and PNNI Node Identifier



The following parameters are displayed for each node. The right column shows the label for each value that appears in the **dsppnni-node** command. The left column maps each value to the corresponding keyword in the **cnfpnni-node** command, and explains the argument function.

node index	The *node-index* is a numeric node identifier assigned by the software—it is not user-configurable. Range: 1–10 In the current release, the only *node-index is* “1,” so the display always shows “1.”
node name	Display the PNNI node name assigned to a PNNI node. Each node name must be unique in the peer group–choose names that all fit into an obviously similar group, for example: names of states, names of universities, and such.
Level	Display the PNNI hierarchical level by defining the length of the *pg-id* value. For example, the default values 56 indicates that the **pgId** value extends 56 bits from the leftmost bit. Therefore, *pg-id* = 47 01 00 8100 0000. If you specify the value of *level* to 40, *pg-id* = 47 01 00 8100.
Lowest	This field indicates whether the node is the lowest logical node on the switch. In a single-peer group, “Lowest” is always true. In a multi-peer group, it can be true or false.

Restricted transit	Display the flag that indicates whether transit is restricted across this node. This value can be set to off to secure the node, or to minimize traffic handled by the node if it is of either low-capacity or high-criticality. on: Calls can transit across this node. off: Only calls terminating on end-systems supported by this node can access this node.
Complex node	Indicates whether this node is a complex node. A node at the lowest level cannot be a complex node. Therefore, in the current release, this field always contains "off." on: This node is a complex node. off: This node is not a complex node.
Branching restricted	Indicates whether the node supports point-to-multipoint branching. on: This node does not support point-to-multipoint branching. off: This node supports point-to-multipoint branches.
Admin status	Display the administrative status of the node. You can disable or enable a node by executing the **cnfpnni-node** command with appropriate parameters. up: The logical PNNI node is enabled. down: The logical PNNI node is disabled.
Operational status	Display the operational status of the node. The software determines the operational state, so you cannot configure it.
Non-transit for PGL election	Display whether transit is restricted across this node. This value is set by the software and is not user-configurable. on: Only calls that terminate on this node can access this node. off: Calls can transit this node.
Node id	For the constituents of the node ID, see the description of **addpnni-node** or **cnfpnni-node**. Before you change the node ID, *you must* disable the node by entering **cnfpnni-node -enable false**. See description of **cnfpnni-node**.

ATM address	For the constituents of the ATM address, see the description of **addpnni-node** or **cnfpnni-node**. Before you change the ATM address, disable the nod by executing **cnfpnni-node -enable false**. See description of **cnfpnni-node**
Peer group id	Display the -**pgId** of length *level* that is assigned to the PNNI node. The peer group is the PNNI local group.The peer group consists of all PNNI nodes with matching *pg-id* values. The default value of *level* is 56 (7 bytes), which specifies the length of **-pgId** to 7 bytes. However, the maximum length of **-pgId** is 14 bytes, so display commands always display -**pgId** as 14 bytes with trailing zeros filling the undefined fields. If you increase the value of *level*, you change the length, and therefore the value, of **-pgId**, but it will always be displayed as 14 bytes. This is a 14-byte, formatted hexadecimal string. Like all PNNI addresses, identifiers, and prefixes, this value is portrayed as a string of hexadecimal “nibbles.” One or several pairs of nibbles entail each parameter field. (See Figure 5-7.)

Example

Display details about the current node. This example reflects a node in a single-peer group.

```
SanJose.7.PXM.a > dsppnni-node 1

node index: 1                     node name: SanJose
   Level...............        56     Lowest..............      true
   Restricted transit..       off     Complex node........       off
   Branching restricted        on
   Admin status........        up     Operational status..        up
   Non-transit for PGL election..       off
   Node id...............56:160:47.00918100000000309409f1f1.00309409f1f1.01
   ATM address...........47.00918100000000309409f1f1.00309409f1f1.01
   Peer group id.........56:47.00.9181.0000.0000.0000.0000.00

SanJose.7.PXM.a >
```

dsppnni-node-list

Display PNNI Nodes List—display a list of all learned PNNI nodes in the network.

The **dsppnni-node-list** command lists the PNNI nodes in the network that are visible to the current switch. For a single-peer group (SPG), the displayed nodes exist at one level. For a multi-peer group (MPG), the list contains all nodes on the current switch and all nodes that are visible to every node on the current switch. The display for an MPG shows an ascending order of nodes based on the node number. From the node list, you can create a graphical representation of the network. The display contains the following information for each node:

- The node number: all nodes in the network that are visible to the local node at a given level are identified by a unique number and stored in a list.
- The node ID is the 22 octet that uniquely identifies the node within the routing domain. See **addpnni-node** or **cnfpnni-node** for components of the node ID.
- The node level, also configured through either the **addpnni-node** or the **cnfpnni-node** command.
- Node name (results from **cnfname**).
- In a multi-peer group (MPG), the index number for nodes above the lowest level are added to the node name.

Note This display may not update frequently enough for you if you are configuring the network. You can change timers to update more frequently, but changing timers can have unexpected effects. Before you change any timers, discuss it with the TAC or your Cisco representative. The **dsppnni-link** command frequently updates a display of the address, link, and Hello packet information of each link.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-node-list`

Syntax Description

This command takes no parameters.

Related Commands

addpnni-node, **cnfpnni-node, cnfname, dsppnni-path, dsppnni-reachable-addr**

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Display Contents for the dsppnni-node-list Command

This section describes the contents of a node list. For a multi-peer group (not applicable in the current release), it also describes how the display changes from one level of a hierarchy to a higher level. Although the description relates primarily to the example of a multi-peer network diagram in Figure 5-8, the information also substantially applies to the example SPG that follows the MPG example. The diagram in Figure 5-8 reflects the MPG display in the Example section.

node #	The node number (node #) is a reference to the nodes in the *network*—not a node in the hierarchy of an MPG on a switch (see the **dsppnni-node** description for details about *node index*). The entity that has this view and compiles this list of node numbers is a local logical node. The node numbers have a range of 1–256. Node # 1 is the logical node that is making its list of network nodes. In an MPG list, multiple instances of node # 1 appear because the logical node at each level sees itself as node # 1. Also, each node in a multi-peer group has information for nodes in its peer group but also for all nodes on the level of its parent, grandparent, and so on. See Figure 5-8 and the Example section. Whether a node belongs to a single-peer group or a multi-peer group, each logical node increments node # by 1 according to the sequence that it discovers other nodes. The paragraphs that follow this list give more details about the node number sequence for a multi-peer group. You can only view a node number in applicable displays or provide it as a command parameter. For example, you can provide a node number to the **dsppnni-path** command.
node id	The *node-id* consists of the level, the length of the ATM address, and the ATM address itself.
node name	The name of the *switch* (not the name of a *logical node*). The root of this node name results from the **cnfname** command. If a dash number follows the node number, that number is the node index that pertains to the hierarchy of nodes on the switch. For this command, a number is appended only for nodes above the lowest level on the switch.
level	The level is set through **addpnni-node** or **cnfpnni-node**. It has a range of 1–104 and a default of 56.

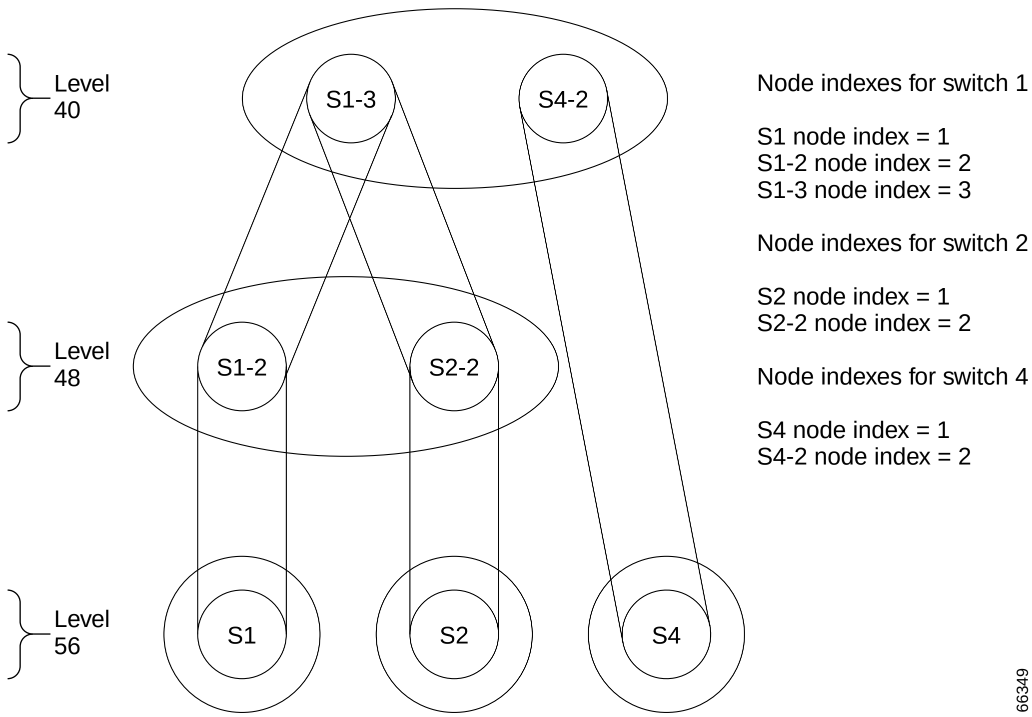
The paragraphs that follow describe the progression in node numbers, levels, and the node index appended to the switch name in an MPG network.Refer to Figure 5-8.

1. The display shows all network nodes that are known to the lowest level. These nodes consist of all nodes in the peer group, all nodes on the level of its parent node, all nodes on the level of its grandparent, and so on. The MPG shown in the Example section illustrates this concept.

   The numbers in the “node #” column begin with 1—which is the local node itself—then continues with the next learned node, and so on. The node# increases by 1 with each discovered node.
2. When the sequence re-starts with node # 1, the display has begun showing the view from the next higher node in the hierarchy. At this next higher level, node # 1 is the current node itself, which has made its own list of nodes. The display continues with peer group members of that level, the members of the parent’s group, the grandparent and members of the grandparent’s group, and so on. The MPG in the Example section illustrates this progression.

3. For levels *above* the lowest, the node index is appended to the name of the switch. See the node name column in the display for a multi-peer group in the Example section. For the definition of a node index, see the description of the **dsppnni-node** command.

Figure 5-8 Multi-Peer Group



Example of MPG

Display all network nodes that are known to the logical nodes on the current switch. This multi-per group is the basis of Figure 5-8. In fact, Figure 5-8 was constructed from this list. Note that node # 2 has been either disabled or deleted from the network.

The first graphical representation in this example is Figure 5-9. It illustrates where the view of a particular level begins and ends. As Figure 5-9 shows, the number of nodes visible at each higher level decreases. After acquiring a visual grasp of the **dsppnni-node-list** display, the screen capture that follows Figure 5-9 provides a more readable list to examine.

The first series of node numbers is node # 1 through node # 6 and is the list complied by the lowest level node. As reflected in Figure 5-8, mpglax1 is the only node in its peer group, so the first series shows only one node at level 56. The remainder of the nodes in the first series are the members at the level of its parent and grandparent, as Figure 5-8 illustrates. At the levels other than the lowest, the node index is appended to the switch name.

For the switch named mpglax4, only levels 56 and 40 were configured. Therefore, the display shows the last node name at level 40 of “mpglax4-02.”

Figure 5-9 A List of Nodes in a Multi-Peer Group

```
mpglax1.1.PXM.a > dsppnni-node-list

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   56:160:47.009181000000003071f80e4a.003071f80e4a.01 mpglax1        56

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   48:56:47.339181000000000000000000.003071f80833.00  mpglax1-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    4   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    5   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    6   48:56:47.119181000000000000000000.003071f80e56.00  mpglax2-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   48:56:47.339181000000000000000000.003071f80833.00  mpglax1-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    2   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   48:56:47.119181000000000000000000.003071f80e56.00  mpglax2-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    4   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40
```

View from mpglax level 56

View from mpglax-02 level 48

View from level 40

66348

```
mpglax1.1.PXM.a > dsppnni-node-list

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   56:160:47.00918100000000003071f80e4a.003071f80e4a.01 mpglax1        56

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   48:56:47.339181000000000000000000.003071f80833.00  mpglax1-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    4   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    5   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    6   48:56:47.119181000000000000000000.003071f80e56.00  mpglax2-02     48

node #  node id                                            node name    level

------- -------------------------------------------------- ---------- -------
    1   48:56:47.339181000000000000000000.003071f80833.00  mpglax1-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    2   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   48:56:47.119181000000000000000000.003071f80e56.00  mpglax2-02     48

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    4   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   40:48:47.559181000100000000000000.003071f80833.00  mpglax1-03     40

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3   40:56:47.229181000000000000000000.003071f80e52.00  mpglax4-02     40

mpglax1.1.PXM.a >
```

Example of SPG

Display the node list for a single-peer group. Note that the level for each node is 56.

```
Geneva.7.PXM.a > dsppnni-node-list

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    1   56:160:47.00918100000001029300121.001029300121.01 pswpop6        56

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    2   56:160:47.009181000000000c043002de1.00c043002de1.01 pswpop7        56

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    3 56:160:47.00918100000000000000000.001029300121.00  pswpop6-02     56

node #  node id                                            node name    level
------- -------------------------------------------------- ---------- -------
    4   56:160:47.009181000000000500ffde80b.00500ffde80b.01 orses18        56

Geneva.7.PXM.a >
```

dsppnni-path

Display PNNI Path—display the pre-computed path tables.

The **dsppnni-path** command displays the pre-computed paths between the source (the current node) and the destination nodes. The system has determined these paths to be the best or optimal paths for various service classes according to one of three metrics. These metrics are the administrative weight (AW), the cell transfer delay (CTD), or the cell delay variation (CDV). The service class that each metric supports for the purpose of routing varies. For a list of metrics and service classes, see Table 5-3. For **dsppnni-path**, you must specify a combinations of routing metric and service class.

Table 5-3 Routing Criteria and Service Classes

Routing Metric	Applicable Service Classes
AW	CBR, ABR, UBR, rt-VBR, nrt-VBR
CTD	CBR, rt-VBR, nrt-VBR
CDV	CBR, rt-VBR

You can also specify a node index. If you specify a *node-index*, the command displays the paths from the source node to the node whose index you specify. To see a list of node indexes, execute **dsppnni-node-list**. (The **dsppnni-node-list** command displays network-level node indexes under the heading "node #.")

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-path {aw {cbr|rtvbr|nrtvbr|ubr|abr} | ctd {cbr|rtvbr|nrtvbr} | cdv {cbr|rtbvr}}
[node-index]
```

Syntax Description

Note The mandatory keywords in this command do not take the usual dash that many keywords require. If you include a dash in front of the keyword, the system rejects the command.

Note The optional *node-index* lets you select a specific node in the network whose path connectivity you want to see. The local node creates the node indexes (or node numbers) according to the sequence that it discovers its neighbors. You can only provide it as a command parameter or view it in applicable displays. Whether or not you specify *node-index*, the node indexes appear in the "node#" column. (Note that this node index or node# is not the node index that identifies a node within the hierarchy of a multiple-peer group. See **dsppnni-node** for details on the local node index.)

aw	Specify administrative weight as the routing metric. The possible service classes associated with AW are CBR, rt-VBR, nrtVBR, and UBR.
ctd	Specify cell transfer delay as the metric. The possible service classes are CBR, rt-VBR, or nrt-VBR.
cdv	Specify cell delay variation as the metric. The possible service classes are CBR and rt-VBR.
node-index	The node index is a number in the range 1–256 that uniquely identifies a switch within a PNNI network. This option lets you specify one destination switch to show connecting paths, otherwise the paths to all switches appear in the display. Range: 1–256 Default: 1

Display Contents

S or D	The S or D in the first column of the display shows whether the line pertains to the source (S) or begins one or more lines about the destination (D).
node #	The node number (node index) within the network. This node number is a unique identifier of the node within the network and appears as “node-index” in many displays. Range: 1–256
PortId	The PNNI logical port identifier in the form of a 32-bit number. Certain commands require the port ID in this format. To obtain the logical port ID from the physical port ID, use the **dsppnports** command. Range: 1–2147483648
node id	The node identifier (node ID) assigned to a PNNI node. The commands that specify the node ID are **addpnni-node** and **cnfpnni-node**. To see the node ID, use **dsppnni-node**.
node name	The name of the switch assigned by the **cnfname** command. This name appears in the CLI prompt.

Related Commands

None

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

The routing criteria for both examples is UBR service with routing metric AW. First, display the pre-computed paths between the current source and the node with index number 5. Thereafter, enter **dsppnni-path** but do not include an index number so the display shows all paths.

```
SanJose.7.PXM.a > dsppnni-path aw ubr 5
node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045505 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045506 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045507 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045508 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose
```

The example for all paths in the network with AW as the routing metric and UBR as the service class, the display is very large and so is truncated. Note that this display shows multiple paths to the source.

```
SanJose.7.PXM.a > dsppnni-path aw ubr
node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  2/         0 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  3/         0 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  3/         0 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  4/         0 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  17111041 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  4/         0 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  16848897 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose
```

```
node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  4/         0 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  16848897 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045505 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045506 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045507 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  5/         0 56:160:47.00918100000000309409f160.00309409f160.01 Chicago
S  1/  17045508 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose


node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  6/         0 56:160:47.00918100000000309409f2a3.00309409f2a3.01 Paris
   4/  17438721 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  17111041 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  6/         0 56:160:47.00918100000000309409f2a3.00309409f2a3.01 Paris
   4/  17438721 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  17111041 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  6/         0 56:160:47.00918100000000309409f2a3.00309409f2a3.01 Paris
   4/  17438721 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  16848897 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                              node name
--------------- ---------------------------------------------------- ----------
D  6/         0 56:160:47.00918100000000309409f2a3.00309409f2a3.01 Paris
   4/  17438721 56:160:47.00918100000000309409f23c.00309409f23c.01 London
   3/  16848897 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose
```

```
node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  7/         0 56:160:47.00918100000000001a531c01.00001a531c01.01 LA
  11/  16848918 56:160:47.00918100000000001a531c83.00001a531c83.01 Jup-1
   3/  16848917 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  7/         0 56:160:47.00918100000000001a531c01.00001a531c01.01 LA
  11/  16848918 56:160:47.00918100000000001a531c83.00001a531c83.01 Jup-1
   3/  16848917 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17438721 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose

node #/PortId   node id                                            node name
--------------- -------------------------------------------------- ----------
D  8/         0 56:160:47.00918100000000309409f213.00309409f213.01 A4b
   7/  16848897 56:160:47.00918100000000001a531c01.00001a531c01.01 LA
  11/  16848918 56:160:47.00918100000000001a531c83.00001a531c83.01 Jup-1
   3/  16848917 56:160:47.00918100000000301a431c19.00301a431c19.01 Boston
   2/  17176577 56:160:47.00918100000000309409f2aa.00309409f2aa.01 Toroton
S  1/  17504257 56:160:47.00918100000000309409f1f1.00309409f1f1.01 SanJose


SanJose.7.PXM.a >
```

dsppnni-pkttrace

Display Packet Trace—display the parameters of a particular packet trace configuration.

This command applies to debugging only.

The **dsppnni-pkttrace** command displays the packet-trace settings. These settings are configured by the **cnfpnni-pkttrace** command. You can use a packet trace to examine the contents of the PNNI Hello packets that are exchanged between two neighboring peers.

Note This command is very intrusive. If you execute it while the node carries live traffic, Cisco recommends that you specify one direction at a time for the trace.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-pkttrace <rx | tx>
[node-index [ -portId port-id | -svcIndex svc-index]]
```

Syntax Description

tx | **rx**	Select a direction for the trace to display. **tx**: transmit **rx**: receive
node-index	In the current release, the only supported value for *node-index* 1. The node index indicates the relative level of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1.) Range: 1–10 Default: 1
-portId	The port ID in this instance has the format of the logical ID number. The format is a 32-bit encoded number in the range 1–2147483648. If you do not have the port ID in this form, use the **dsppnport** command and provide it with the common portID format of *slot*[:*subslot*].*port*[:*subport*}. The output of **dsppnport** shows the logical number for the port ID. Use this value is for the **-portID** parameter.
-svcIndex	An index of the switched virtual connection routing control channel (SVCC-RCC) packet trace. This parameter is meaningful only if you specify *node-index*. Default: none **Note** The current release does not support Routing Control Channels for Switched Virtual Connections (SVCC-RCC), so this value must remain **0**.

Related Commands

cnfpnni-pkttrace

Attributes

Log: nolog	State: active	Privilege: CISCO_GP

Example

First, configure the following packet trace parameters through **cnfpnni-pkttrace**:

- The direction is transmit.
- The node index is 1.
- The logical port identifier is 17373186. If you do not have the logical port identifier, use the **dsppnport** command to see the logical port number derived from the physical port number.

Next, check the packet trace you have configure by executing **dsppnni-pkttrace**.

```
Geneva.7.PXM.a > cnfpnni-pkttrace -tx 17373186
PNNI/tx_packet on port 17373186 at level 56
> 01:00010064  01010100  000038a0  47009181  00000000  309409f3  b8003094
> 02:09f3b801  47009181  00000000  309409f3  b8003094  09f3b801  38470091
> 03:81000000  00000000  000038a0  47009181  00000000  001a531c  2a00001a
.
.
.
Geneva.7.PXM.a > dsppnni-pkttrace tx 1 -portId 17373186

Node Index :1   Port id:        17504   Tx Pkt Trace on

Geneva.7.PXM.a >
```

dsppnni-ptse

Display PNNI topology state element—displays PTSE tables.

The **dsppnni-ptse** command displays PNNI topology state elements (PTSEs). The purpose of this command is troubleshooting, and it requires familiarity with the ATM Forum PNNI 1.0 specification. Without knowledge of this specification, the usefulness of **dsppnni-ptse** is minimal.

PTSE Types

A node indicates its characteristics (such as all its ATM addresses) to all other nodes in the peer group by broadcasting numerous PTSEs. A node periodically sends (or *floods* the group with) PTSEs according to a user-specified timer but also floods the group with PTSEs when it triggers a change of topology. A typical topology change is an addition of an ATM address.

Each PTSE carries an indicator of what type of PTSE it is. This PTSE type appears as both a descriptive string and a number set by the ATM Forum. The section, “Display Contents for dsppnni-ptse” lists the contents of each information group identified by the PTSE type. Five basic types exist, and various subtypes exist. The basic types of PTSEs are:

1. Nodal information group
2. Internal reachable addresses
3. External reachable addresses
4. Horizontal links
5. Uplinks (multiple peer groups only)

Granularity of the Output

The optional parameters let you determine the granularity of the target of the command. The granularity ranges from the whole peer group to a specific logical port. Additionally, you can specify a “detailed” display or just the header information for PTSEs.

If you specify:

- Nothing, the display contains header information for all PTSEs for all logical nodes in the network.
- Only the *node-index*, the output contains all PTSEs sent from the node indicated by *node-index*.
- Only *node-index*, *node-id*, and *ptse-id*, the display shows the PTSE uniquely identified by these three parameters.
- A detailed display, the display contains information about the header and the contents of the PTSE and applies to all combinations of the other parameters.

Note The *node-index* is automatically generated. See description of **dsppnni-node-list**.

The *ptse-id* is generated by the node that sends the PTSE.

You can use **dsppnni-ptse** to trouble-shoot a faulty designated transit list (DTL). If a DTL is faulty, you can observe the PTSE of nodes on the designated path to confirm the accuracy of the information used to build the DTL. You can also use **dsppnni-ptse** to determine if nodes are correctly passing both the topology packets and the Hello packets.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-ptse
[node-index [node-id [ptse-id]]]
[-detail {true | false}]
```

Syntax Description

Note The parameters *node-index*, *node-id*, and *ptse-id* are nested. Therefore, you cannot enter *node-id* without *node-index*, nor can you enter *ptse-id* without *node-index* and *node-id*.

node-index	A unique, network-wide node identifier. This system-generated number has a range of 1–256. Range: 1–256. Default: (no default)
node-id	The user-specified node ID. See **addpnni-node** or **cnfpnni-node** for a description. Default: (no specific node ID)
ptse-id	An integer that identifies a PTSE generated by a particular node. Regardless of the number of times a node sends a PTSE, this ID remains the same until a change to the topology occurs. For example, adding a ATM address to a node causes that node to generate a new PTSE and associated ID. The PTSE ID has a theoretical limit of a 32 bit number. However, the PTSE ID is likely to be a relatively small number.
-detail	Selects the amount of detail for the display. true: Display the contents of the PTSE as well as the header information. false: Display only the PTSE header. Default: false

Display Contents for dsppnni-ptse

This section describes basic information for each PTSE type. In addition, each variation of the command output contains the following header information.

node-index	This unique, network-wide node identifier is a switch-generated number in the range 1–256. If the network consists of a multi-peer group, the display shows the sequence of node numbers for the lowest level then starts the sequence at the next level.
originating node ID	The identifier of the node that broadcast the PTSE.

PTSE ID	The unique identifier of the PTSE. *ptse-id* is a 32 bit number index assigned by the PNNI node that created the PTSE.
PTSE type	The type of PTSE is an ASCII designated by the ATM forum PNNI standard. Broad and narrow categories.

Nodal State Parameter Information

internal reachable ATM addresses	A list of reachable ATM addresses that are inside the peer group or network.
exterior reachable ATM addresses	A list of reachable ATM addresses that are outside the network. The current release does not support exterior addresses.
PTSE length	The number of bytes in the PTSE—a 16-bit number.
sequence	The sequence of the PTSE—a 32-bit number.
checksum	The checksum error-checking value. A 16-bit hex number.
remaining lifetime	The length of the remaining lifetime (in seconds). 32-bit number.
details for IG	The flag that determines the level of details for information group (IG) if the **detail** option is enabled ("true").

Nodal Information Group Parameters

Note The current release does not support the logical group node (LGN), preferred peer group leader (PGL), and other items that pertain to the levels of a multi-peer group.

type	The type of nodal information group (IG).
length	The length of the nodal IG PTSE. A 16-bit number.
ATM address	The upnode ATM address is a 20-byte, hexadecimal string. The upnode is the node at the other end of the uplink. It is the neighboring peer of the ancestor of the node from which the uplink originates.
priority	The value of the priority parameter, an 8-bit number.
nodal flags	The 8-bit nodal flags.
preferred PGL	A 22-byte hex string. The current release does not support peer group leader (PGL).
next higher level binding information IG type	The next higher level binding information IG type is an ASCII string. The current release does not support higher level binding.

next higher level binding information IG length	A 16-bit number. The current release does not support higher level binding.
parent LGN id	The parent LGN ID is a 22-byte hex string. The current release does not support logical group numbers (LGNs).
parent LGN ATM address	The parent LGN ATM address is a 20-byte, hex string. The current release does not support group numbers (LGNs).
parent PG id	The peer group ID (of length *level*) assigned to the parent PG. The peer group is the PNNI local group. The peer group consists of all PNNI nodes with matching **-pgId** values. The current release does not support parent peer groups. Default: Figure 5-1 shows the factory-set default.
parent peer group PGL	The parent peer group PGL identifier is a 22-byte hexadecimal string. The current release does not support PGL.

Nodal State IG Parameters

type	The ASCII string that indicates the type of the IG nodal state parameters.
length	A 16-bit number.
flags	A string of 8-bit flags.
input port id	The logical identifier on the input interface. For more details, see PNNI Format. Range: 1–2147483648
output port id	The logical PNNI identifier on the output interface. For more details, see PNNI Format. Range: 1–2147483648

Internal Reachable ATM Address IG Parameters

type	The ASCII string that indicates the type of the internal reachable ATM address IG parameters.
length	A 16-bit number.
flags	A string of 8-bit flags.
port id	The logical PNNI identifier on the interface. For more details, see PNNI Format. Range: 1–2147483648

scope	The UNI 4.0 address scope. **Note** The current release does not support UNI 4.0, so the scope is '0'.
address info length	The length of the address information—an eight-bit number.
address count	The number of reachable addresses—a 16-bit number.
reachable address prefixes	Display any PNNI summary address reachable by the node. The length of addressprefix is set by *prefixlength*.

External Reachable ATM Address IG Parameters

> **Note** The current release does not support exterior reachable addresses.

type	The ASCII string that indicates the type of the exterior reachable ATM address IG parameters.
length	A 16-bit number.
flags	A string of 8-bit flags.
port id	The logical port number of the PNNI port. This format is a 32-bit number. Range: 1–2147483648
scope	An 8-bit number.
address info length	An 8-bit number. The current release does not support exterior reachable addresses.
address count	A 16-bit number.
reachable address prefixes	Display any exterior PNNI summary address reachable by the node. The length of addressprefix is set by *prefixlength*.

Horizontal Links IG Parameters

type	The ASCII string that indicates the type of the horizontal link IG parameters.
length	A 16-bit number.
flags	A string of 8-bit flags.

remote node id	The node ID of the remote node. For a description of the node ID, see the description for **addpnni-node** or **cnfpnni-node**.
remote port id	The logical PNNI identifier on the remote interface. For more details, see PNNI Format. Range: 1–2147483648
local port id	The logical PNNI identifier on the local interface. For more details, see PNNI Format. Range: 1–2147483648
aggregation token	See the description of the **cnfpnni-intf** command for a definition of an aggregation token. The current release does not support link aggregation. Range: 1–32

Uplink IG Parameters

Note The current release does not support uplinks.

type	The ASCII string that indicates the type of up link IG parameters.
length	A 16-bit number.
flags	A string of 8-bit flags.
remote higher level node id	The PNNI node identifier assigned to a PNNI node.
common pg id	The peer group ID (of length *level*) that assigned to the PNNI common PG. This peer group is the local peer group. Default: Figure 5-1 shows the factory-set default.
local port id	The logical PNNI identifier on the interface. For more details, see PNNI Format. Range: 1–2147483648
aggregation token	Range: 1–2147483648.
upnode ATM address	The ATM address of the PNNI uplink node. The upnode ATM address is a 20-byte, hexadecimal string. The upnode is the node at the other end of the uplink. It is the neighboring peer of the ancestor of the node from which the uplink originates. Default: none

Resource Availability IG Parameters

type	Indication of bi-directional resource availability information group (RAIG)
length	A 16-bit number.
flags	A 16-bit number.
aw	The bandwidth used by AW metric cells in cells per second. Range: 1–2147483648.
ctd	The bandwidth used by CTD metric cells in cells per second. Range: 1–2147483648.
cdv	The bandwidth used by CDV metric cells in cells per second. Range: 1–2147483648.
mcr	The bandwidth used by MCR metric cells in cells per second. Range: 1–2147483648.
acr	The bandwidth used by ACR metric cells in cells per second. Range: 1–2147483648.
clr0	The bandwidth used by CLR0 metric cells in cells per second. Range: 1–2147483648.
clr0+1	The bandwidth used by CLR0+1 metric cells in cells per second. Range: 1–2147483648.

Generic Connection Admission Control (GCAC) IG

type	The ASCII string that indicates the type of GCAC IG parameters.
length	A 16-bit number.

crm	The cell rate margin (CRM) is a measure of the difference between the effective bandwidth allocation and the allocation for sustainable cell rate. It is a safety margin allocated above the aggregate sustainable cell rate for nrt-VBR and rt-VBR. This feature has little impact on traffic management. The ATM Forum does not require support for CRM, and Cisco Systems currently does not support it on the Cisco MGX 8850 and SES products. Range: 1–2147483648.
vf	The variance factor (VF) is a relative measure of the square of the cell rate margin (CRM) normalized by the variance of the sum of the cell rates of all existing connections. VF applies to nrt-VBR and rt-VBR, but it has little impact on traffic management. The ATM Forum does not require support for VF, and Cisco Systems currently does not support it on the Cisco MGX 8850 and SES products. Range: 1–2147483648.

Related Commands

None

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

For the first example, enter the command with no parameters, so all information about every node on the switch appears.

```
Geneva.7.PXM.a > dsppnni-ptse -detail true
node index: 1
originating node name: Krishna
originating node id: 56:160:47.0091810000000000c0326496.0000c0326496.00
Type................ 64 Length............. 1200
Sequence number..... 1 Checksum............ 94d
PTSE id............. 1 Remaining lifetime.. 2997
PTSE type........... Nodal Info( 97)
Type................ 97 Length............. 65
Priority............ 0 Flags.............. f8
ATM addr..............47.0091810000000000c0326496.0000c0326496.00
Pref PGL id...........0:0:00.000000000000000000000000.000000000000.00
binding info: Type 192, Length 76
next level LGN node id. 48:56:47.009181000000000000000000.0000c0326496.00
next level LGN ATM addr 47.0091810000000000c0326496.0000c0326496.30
next level LGN PG id... 30:47.00.9181.0000.0000.0000.0000.00
next level LGN PGL id.. 30:48:56:47.009181000000000000000000.0000c0326496.00
node index: 1
originating node name: Liz
originating node id: 56:160:47.0091810000000000c0326496.0000c0326496.00
```

```
Type................ 64 Length.............. 1200
Sequence number..... 1 Checksum............ 94d
PTSE id............. 2 Remaining lifetime.. 2997
PTSE type........... Nodal State Parameter( 96)
Type................ 96 Length.............. 65
Reserved............ 0 Flags............... 0
Input port id....... 48 Output port id...... 12
```

For the second example, specify the following parameters:

- The *node-index* is 1.
- The *node-id* is 56:160:47.00918100000000107b65f27c.00107b65f27c.01 .
- The PTSE ID is 28.

Display the PTSEs for node index 2. After listing the PTSEs, display details for one particular PTSE—19 in this example.

Note The presence of the colons in the node ID are required, but the periods are optional.

```
M8850_NY.7.PXM.a > dsppnni-ptse 2

node index: 2
originating node name: M8850_NY-02
originating node id: 48:56:47.00918100000200000000000.00036b5e30cd.00
   Type................        64     Length..............        96
   Sequence number.....       155     Checksum............      689b
   PTSE id.............         1     Remaining lifetime..      3224
   PTSE type...........  Nodal Info( 97)

node index: 2
originating node name: M8850_NY-02
originating node id: 48:56:47.00918100000200000000000.00036b5e30cd.00
   Type................        64     Length..............        44
   Sequence number.....       153     Checksum............      d7fd
   PTSE id.............        18     Remaining lifetime..      3224
   PTSE type...........  Int Reach Addr(224)

node index: 2
originating node name: M8850_NY-02
originating node id: 48:56:47.00918100000200000000000.00036b5e30cd.00

Type................        64     Length..............        52
   Sequence number.....       152     Checksum............       bba
   PTSE id.............        19     Remaining lifetime..      3224
   PTSE type...........  Int Reach Addr(224)
```

As directed at the beginning of this example, display details for PTSE 19.

```
M8850_NY.7.PXM.a > dsppnni-ptse 2 48:56:47009181000002000000000000000036b5e30cd00 19
-detail true

node index: 2
originating node name: M8850_NY-02
originating node id: 48:56:47.00918100000200000000000000.00036b5e30cd.00
   Type................       64     Length..............       52
   Sequence number.....      152     Checksum............      bba
   PTSE id.............       19     Remaining lifetime..     2389
   PTSE type...........  Int Reach Addr(224)

   Type................      224     Length................        32
   Reserved............        0     Flags.................      8000
   Port id.............        1     Scope.................         0
   Ail.................       14     Aic...................         1
   prefix..............47.0091.8100.0000.0003.6b5e.30cd./104
```

dsppnni-reachable-addr

Display PNNI Reachable Addresses—displays the reachable PNNI addresses in the peer group.

This command displays all the reachable addresses and address prefixes in the peer group. For a description of the items in the display, refer to the section, “Display Contents for dsppnni-reachable-addr.” The display granularity depends on your parameter choice:

- If you enter **local**, the display shows the port ID and the addresses directly attached to the local node.
- If you enter **network**, the display shows the advertising node ID, the addresses advertised by other nodes, and the routing parameters for each reachable node.

Note The display may not update frequently enough if you are configuring the network. You can change timers to update more frequently, but changing timers can have unexpected effects. Before you modify a timer, discuss it with the TAC or your Cisco representative.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-reachable-addr <local | network>
```

Syntax Description

local | network	Ddetermine whether the display shows the addresses of nodes that directly connect to this switch or all nodes in the peer group. Local: directly connected switches Network: all reachable nodes in the peer group Default: (no default)

Display Contents for dsppnni-reachable-addr

The table contains all reachable addresses within a peer group comes from the internal data base (IDB).

scope	**Note** The UNI 4.0 address scope. Refer to ATM forum documentation for a description of these scopes. The current release does not support UNI 4.0, so *scope* currently is always ‘0’. Range
port id	The logical port identifier.
Exterior	The flag that indicates whether the node is an interior or exterior node. true: the node is an exterior node. false: the node is an interior node.
ATM addr prefix	The PNNI summary address assigned to the node.
node name	The name of the switch results from the **cnfname** command and appears in the CLI prompt.
Advertising node number	The number of the remote node that has advertised information to the current node. This number has a range of 1–256 and appears only if you specified the **network** argument. The local node generates the node numbers in the sequence that it discovers its neighbors. You can only provide it as a command parameter or view it in applicable displays. (Note that this node index or node number is not the node index that identifies a node within the hierarchy of a multiple-peer group. See **dsppnni-node** for details on the local node index.)
Transit Network ID	The transit network ID identifies a network where connections from the current node do not terminate.This number applies to static addresses only. The application of this option depends on the design intent of the user. The ID can have up to four IA5 characters (IA5 is a superset of the ASCII character set).

Related Commands

dsppnni-link

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the reachable addresses that directly connect to this node: the parameter is **local**.

```
Geneva.7.PXM.a > dsppnni-reachable-addr local

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010b.180b/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010b.1816/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010b.1820/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010b.1821/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010d.1820/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010d.1821/152

scope...............        0     port id............4294967295
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0030.ff0f.ef38.0000.010d.1822/152
```

Display all the addresses and address prefixes that are reachable from this node, and display the routing parameters for each reachable node in each direction.

```
Geneva.7.PXM.a > dsppnni-reachable-addr network

scope...............        0     Advertising node number        13
Exterior............    false
ATM addr prefix.....47.0091.8100.0000.0010.7b65.f27c/104
Advertising nodeid..56:160:47.00918100000000309409f13f.00309409f13f.01
Node name...........Moscow
                        forward direction
            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MaxCR    351500   351500   351500   351500   351500
AvCR     290935   290935   290935   290935   290935
CTD          41       41       41      n/a      n/a
CDV          10       10      n/a      n/a      n/a
CLR0         10        8        6      n/a      n/a
CLR0+1        8        8        8      n/a      n/a
CRM         n/a      n/a      n/a      n/a      n/a
VF          n/a      n/a      n/a      n/a      n/a
```

```
                        backward direction
            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MaxCR    351500   351500   351500   351500   351500
AvCR     290935   290935   290935   290935   290935
CTD          41       41       41      n/a      n/a
CDV          10       10      n/a      n/a      n/a
CLR0         10        8        6      n/a      n/a
CLR0+1        8        8        8      n/a      n/a
CRM         n/a      n/a      n/a      n/a      n/a
VF          n/a      n/a      n/a      n/a      n/a

scope...............         0     Advertising node number         8
Exterior............     false
ATM addr prefix.....47.0091.8100.0000.0010.7b65.f27c/104
Advertising nodeid..56:160:47.00918100000000107b65f27c.00107b65f27c.01
Node name...........Paris


scope...............         0     Advertising node number         8
Exterior............      true
ATM addr prefix.....47.0091.8100.0000.0030.9409.f13f/104
Advertising nodeid..56:160:47.00918100000000107b65f27c.00107b65f27c.01
Node name...........Paris

                        forward direction
            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MaxCR    351500   351500   351500   351500   351500
AvCR     290935   290935   290935   290935   290935
CTD          41       41       41      n/a      n/a
CDV          10       10      n/a      n/a      n/a
CLR0         10        8        6      n/a      n/a
CLR0+1        8        8        8      n/a      n/a
CRM         n/a      n/a      n/a      n/a      n/a
VF          n/a      n/a      n/a      n/a      n/a

                        backward direction
            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MaxCR    351500   351500   351500   351500   351500
AvCR     290935   290935   290935   290935   290935
CTD          41       41       41      n/a      n/a
CDV          10       10      n/a      n/a      n/a
CLR0         10        8        6      n/a      n/a
CLR0+1        8        8        8      n/a      n/a
CRM         n/a      n/a      n/a      n/a      n/a
VF          n/a      n/a      n/a      n/a      n/a

Geneva.7.PXM.a >
```

dsppnni-routing-policy

Display PNNI Routing Policy—display the PNNI routing policy parameters.

The **dsppnni-routing-policy** command displays the parameters associated with the current routing policy for this node. The displayed parameters determine:

- The tolerance of cost-calculations.
- The frequency of routing table generation.
- The type of load balancing that is specified.
- The type of on-demand routing that is specified.
- The type of administration weight table that is enabled.

Caution You can change the routing policies to optimize PNNI routing for your network, but incorrect routing policies can cripple or even crash a network. You should not change routing policies on a live network. Use this command only after careful planning.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-routing-policy
```

Display Contents

This section lists the displayed information for each node. The display shows the configuration that results from **cnfpnni-routing-policy**.

SPT epsilon	This parameter is meaningful primarily for crankback. The *epsilon* you supply specifies a tolerance in the form of a percent that can influence which paths qualify as equal-cost during route calculation. A higher tolerance results in a broader range of path cost-values that can qualify as equal-cost. If two paths have very similar administrative weights (AWs), a large enough tolerance eliminates equal-cost as a routing factor because the routing algorithm regards the costs as equal. The range of 0–20 for this parameter comes from the ATM Forum PNNI specification. However, the percent of tolerance that the numbers dictate is determined by individual vendors. Cisco Systems currently maps the following percentages for the Cisco MGX 8850 switch: 0: the total AWs along both directions of the path must be identical. 1-2: the total AWs along both directions of the path must be within 1.06% 3-4: the total AWs along both directions of the path must be within 3.125% 5-9: the total AWs along both directions of the path must be within 6.25% 10-15: the total AWs along both directions of the path must be within 12.5% 16-20: the total AWs along both directions of the path must be within 25.0% Range: 0–20 Default: 0, so only identical path-cost values qualify as equal-cost
Load balance	The load balancing rule if any alternative, equal-cost routes exist for a given call request. random = requires the least overhead due to minimal calculation. Best choice when selecting between paths that have similar available bandwidth. maxbw = requires the most overhead due to ongoing comparison of available bandwidth on paths. Best choice when selecting between paths with dissimilar or fluctuating bandwidth.
SPT holddown time	Minimum time between consecutive calculations that generate routing tables. Units: 100 millisecond increments Range: 1–600 (0.1–60 seconds)

On demand routing	The current rule for on-demand routing is *firstfit* or *bestfit*. The firstfit routing policy selects the first route found that goes to the destination. The time for finding a route is the least possible, but the optimal route may not be selected. The bestfit policy selects a route based on: • The least-cost route, where the sum of all administrative weights in both directions of the route must be less than *maxCost*. • Link verification. • Path constraint checks. • Avoidance of blocked nodes and links. • Checking limits in the designated transit list (DTL).
SPT path holddown time	The minimum time that can elapse between consecutive calculations that generate routing tables on border nodes. Units: 100 milliseconds. Range: 1 (0.1 seconds)–600 (60 seconds). **Note** The current release does not support border nodes, so this value must remain off.
AW Back-ground Table	The flag that enables or disables administrative weight (AW) for the background routing table. The AW is the cost to traffic that traverses that path. The metric AW can be specified on the interface and by the service class (or QoS class), and it is associated with each link. AW is a defining factor when routes are selected. The AW parameters influence how PNNI selects paths in the peer group, and therefore how it distributes each SVC and SPVC. PNNI route selection can also key on AW to exclude certain links from routing, such as defining a backup link for use only when there is no available bandwidth on the primary link. The AW for a path is the sum of all AWs at each port egress for both directions on the path.
CTD Back-ground Table	The flag that enables or disables cell transfer delay (CTD) for the background routing table. CTD is the time interval between a cell exiting source node and entering the destination node.
CDV Back-ground Table	The flag that enables or disables cell delay variation (CDV) for the background routing table. CDV is a component of cell transfer delay, and is a quality of service (QoS) delay parameter associated with CBR and VBR service. Cell Delay Variation is the variation of delay between cells, measured peak to peak.

Related Commands

cnfpnni-routing-policy

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the parameters associated with the current routing policy for this node.

```
Geneva.7.PXM.a > dsppnni-routing-policy

   SPT epsilon.........         0     Load balance........    random
   SPT holddown time...         1     On demand routing...  best fit
   SPT path holddown time       2     AW Background Table         on
   CTD Background Table        on     CDV Background Table        on

Geneva.7.PXM.a >
```

dsppnni-scope-map

Display Scope Map—display the PNNI scope map table.

The **dsppnni-scope-map** command displays the table that maps UNI 4.0 scope to PNNI hierarchy level.

Note The scope map table requires UNI 4.0 support, so this command is not useful in the current release.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-scope-map`

Objects Displayed:

Displays the entire scope map table.

Related Commands

cnfpnni-scope-map

Attributes

Log: nolog	State: active	Privilege: ANYUSER

Example

This command is not supported by the current release.

dsppnni-spoke

Display PNNI Spoke—display the PNNI complex node default advertisement values.

Note This debugging command does not apply to single-peer groups.

The **dsppnni-spoke** command displays how the UNI 4.0 address scope values map to the PNNI hierarchal levels. It displays the PNNI default spoke for a logical group node (LGN) using complex node representation in a given peer group (PG). The spoke is the conceptual "radius" of the peer group. The spoke values are based on averaging the administrate weights (AWs) of all border node paths then dividing that average by 2.

If a logical path is not included in the bypass table, the spoke values can be used to select which peer group a route transits. The PG with the lowest spoke AW is the lowest cost PG and therefore the best path to use (based on AW).

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-spoke` `<`*`node-id`*`>`

Syntax Description

node-id	The node identifier of a PNNI logical node can be user-assigned by **addpnni-node** or **cnfpnni-node** but also comes as a factory-assigned default. Default: (the factory-set default)

Display Contents

The following parameters are displayed for each node.

nodal aggregation method	The ASCII string of the active aggregation method. The method is full-meshed or spanning tree.
ptse-id	The unique identifier for the PTSE. *ptse-id* is assigned by the PNNI node that created the PTSE.
node-index	The *node-index* is the local node index and has a range of 1–10. Range: 1–10
AW-NRTVBR	The administrative weight for nrt-VBR connections on this interface. Range: 0–4194304

AW-CBR	The administrative weight for CBR connections on this interface. While a CBR connection is active, this option limits its bit rate to a static value that remains available until the connection is torn down. The bit rate is characterized by the peak cell rate (PCR) value. Range: 0–4194304
AW-ABR	The administrative weight for available bit rate (ABR) connections on this interface. Specify the 24 bit number AW for ABR on this interface. Range: 0–4194304
AW-RTVBR	The administrative weight for rt-VBR connections on this interface. Range: 0–4194304
AW-UBR	The administrative weight used for unspecified bit rate (UBR) connections. This category includes switched virtual connection (SVC) ping connections. Range: 0–4194304

Related Commands

None

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the spoke mapping for LGN 56:160:47.00918100000000309409f1f1.00309409f1f1.0.

```
SanJose.7.PXM.a > dsppnni-spoke 56:160:47.00918100000000309409f1f1.00309409f1f1.0
node index: 1
  Ptse id ............      948      Flags................ a3
  Nodal aggregation method.. spanning tree

            CBR    RTVBR   NRTVBR      ABR      UBR
         ------   ------   ------   ------   ------
AW         5040     5040     5040     5040     5040
MCR           0        0        0        0        0
AvCR     100000   100000   100000   100000   100000
CTD           0        0        0        0        0
CDV           0        0        0        0        0
CLR           0        0        0        0        0 0
CLR0+1        0        0        0        0        0
CRM          10       10       10       10       10
VF            5        5        5        5        5
```

dsppnni-summary-addr

Display PNNI Summary Address—display the PNNI summary addresses.

The **dsppnni-summary-addr** command displays all summary addresses at the specified degree of granularity.

- If you specify *node-index*, the command displays the PNNI summary addresses of the *node-index* PNNI node.
- If you do not specify *node-index*, the command displays PNNI summary addresses for all local nodes on network.

Use **addpnni-summary-addr** to create a new summary addresses or to configure an existing one.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-summary-addr
[node-index]
```

Syntax Description

node-index	Specify the system-generated identifier of a logical node within a hierarchy. In the current release, the only *node-index* is 1. Range: 1–10. Default = 1.

Display Contents

node index	The number of the node within the hierarchy on this switch. The range for a multi-peer group is 1–10. For a single-peer group, the only node index is 1.
Type	Display the value of the argument **-type**—whether the kind of PNNI summary address is internal or external. In the current release, the value of the **-type** argument must be ‘internal’. internal: This PNNI summary address includes only addresses that are within the peer group. exterior: This PNNI summary address includes addresses that are outside of the peer group.
Suppress	Display the value of the argument -**suppress—**whether the node PNNI summary address is advertised or suppressed. false: The PNNI summary address is advertised (is not suppressed). true: The PNNI summary address is not advertised (is suppressed).

State	This system-generated ASCII string indicates the advertisement state. Possible states: “advertising,” “notadvertised,” or “inactive”
Summary address	The ATM PNNI summary address assigned to the network. The default is a combination of the peer group ID appended with the switch MAC address.
prefixlength	The length of the summary *address-prefix* in number of bits, equal or less than 152 bits. In the current release, the zero-length PNNI summary address is not supported.

Related Commands

addpnni-summary-addr, **delpnni-summary-addr**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the PNNI address prefixes. This command line does not specify *node-index*, so the output contains all PNNI summary addresses in the peer group rather than a specific node. In this case, only a single peer group exists.

```
Geneva.7.PXM.a > dsppnni-summary-addr

node index: 1
   Type..............    internal     Suppress..............   false
   State............. advertising
   Summary address........47.0091.8100.0000.0030.ff0f.ef38/104

node index: 1
   Type..............    internal     Suppress..............   false
   State.............    inactive
   Summary address........47.0091.8100.0000.0010.7b65.f260/104

Geneva.7.PXM.a >
```

dsppnni-svcc-rcc

Display PNNI Switched Virtual Connection Routing Control Channel (SVCC-RCC)—display the PNNI SVC-based RCC table.

The **dsppnni-svcc-rcc** command displays the SVCC-RCC connection and packet values.

If you specify:

- Both *node-index* and *svc-index*, the display shows information about an SVCC-based RCC.
- Only *node-index*, the display shows all SVC-based RCCs attached to the *svc-index* node.
- Nothing, the display shows all SVC-based RCCs attached to all PNNI nodes in the network.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnni-svcc-rcc
[node-index [svc-index]]
```

Syntax Description

node-index	This system-generated indicates the relative position of a logical node within a hierarchy. It has a range of 1–10. For a single-peer group, the only value for *node-index* is 1. Range: 1–10 Default: 1
svc-index	PNNI uses the SVC index as a reference to the horizontal link (H-link) between the levels in a multi-peer group. An SVC serves as the connection for an H-link.

Display Contents

The **dsppnni-svcc-rcc** command displays node, Hello packet, and SVC information for each RCC.

Related Commands

None

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

This command is not supported by the current release.

This example shows the **dsppnni-svcc-rcc** command line that displays SVC-based RCCs.

```
Geneva.7.PXM.a > dsppnni-svcc-rcc
node index: 1 svc index: 33
Hello pkt RX........ 34 SVCC VPI........... 34
Hello pkt TX........ 34 SVCC VCI........... 128
Hello state.......... 2wayOutside
Remote node id.........56:160:39.840f80113744000000400202.00107b0efe01.00
Remote node ATM addr...39:840f.8011.3744.0000.0040.0102.4000.0c80.8030.00
node index: 2 svc index: 33
Hello pkt RX........ 34 SVCC VPI........... 34
Hello pkt TX........ 34 SVCC VCI........... 128
Hello state...........2wayOutside
Remote node id.........56:160:39.840f80113744000000400202.00107b0efe01.00
Remote node ATM addr...39:840f.8011.3744.0000.0040.0102.4000.0c80.8030.00

Geneva.7.PXM.a >

mpgses1.2.PXM.a > dsppnni-svcc-rcc

node index: 2                      svc index: 1
   Hello pkt RX........        68     SVCC VPI...........         1
   Hello pkt TX........        67     SVCC VCI...........        35
   Hello state...........twoWayInside
   Remote node id.........48:56:47.00918100000000000000000022.003071f80e56.00
   Remote node ATM addr...47.009181000000003071f80e56.003071f80e56.02


node index: 3                      svc index: 2
   Hello pkt RX........        57     SVCC VPI...........         1
   Hello pkt TX........        54     SVCC VCI...........        36
   Hello state...........twoWayInside
   Remote node id.........40:56:47.00918100000000000000000033.003071f80e52.00
   Remote node ATM addr...47.009181000000003071f80e52.003071f80e52.02
```

dsppnni-svcc-rcc-timer

Display PNNI Switched Virtual Connection Routing Control Channel (SVCC-RCC) Timer Values—display the PNNI SVCC-based RCC timer values.

The **dsppnni-svcc-rcc-timer** command displays the SVCC-RCC timer values that are set by the **cnfpnni-svcc-rcc-timer** command.

Note This command applies to multi-peer groups only.

If you specify *node-index*, the command displays the SVCC-based Routing Control Channel (RCC) timer values of the *node-index* PNNI node.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-svcc-rcc-timer`
`[`*`node-index`*`]`

Syntax Description

node-index	Specify the node identifier in the range 1–10. In the current release, the value of the *node-index* variable must be ‘1’. Range: 1–10 Default: 1

Display Contents

The following parameters are displayed for each node.

node-index	The local node identifier within the hierarchy. The range is 1–10. In the current release, the value of *node-index* is ‘1’. Range: 1–10
Init time	Display the value of **-initTime**—the interval (in sec) that this node delays advertising its choice of a preferred SVCC to a neighbor with a numerically lower ATM address, The interval begins when the SVCC is established. Range: 1–10
Retry time	Displays the interval (in sec) this node will delay after an apparently necessary and viable SVCC-based RCC is unexpectedly torn down, before attempting to re-establish it. Range: 10–60

Calling party integrity time	Display the value of **callingIntegrityTime**, which limits wait times for establishing an SVCC as a called party. After the node has decided to accept an SVCC as the called party, the **calledIntegrityTime** variable specifies the interval (in sec) that this node will wait for an SVCC to become fully established before giving up and tearing down the connection. Range: 5–300
Called party integrity time	Display the value of **calledIntegrityTime**, which limits wait times for establishing an SVCC as a called party. After the node has decided to accept an SVCC as the called party, the **calledIntegrityTime** variable specifies the interval (in sec) that this node will wait for an SVCC to become fully established before giving up and tearing down the connection. Range: 10–300

Related Commands

dsppnni-svcc-rcc-timer

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

dsppnni-timer

Display PNNI Timer—display the PNNI timer values.

The **dsppnni-timer** command displays the nodal timer values configured through the **cnfpnni-timer** command. If you provide a node index with the command, the output contains information for that particular node rather than all logical nodes on the switch.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnni-timer`
[*`node-index`*]

Syntax Description

node-index	The node index indicates the relative position of the logical node within a multi-peer group on the switch. The range is 1–10, and the lowest level is 1. If you do not have the node index, use **dsppnni-node** to see a list of all logical nodes and node indexes on the current switch. In the current release, the value of *node-index* must be ‘1’. Range: 1–10 Default: 1

Display Contents

The display contains the following information for each node. Except for the node index, you can configure all values through the **cnfpnni-time**r command.

node index	The relative position of the local node on the switch.
Hello holddown	The initial value for the Hello hold down timer is the time a node waits to send Hello packets. Units: 100 milliseconds (1 = 0.1 seconds)
PTSE holddown	The time the node waits to broadcast PNNI topology statement elements (PTSEs).
Hello int	The initial time in millisecond-increments that the node uses to limit the rate of at which it transmits Hello packets. Units: 100 milliseconds (1 = 0.1 seconds)
PTSE refresh int	The *initial* number of seconds allowed for the PTSE to re-originate.

Hello inactivity factor	The *Hello inactivity factor* figures in the generation of a time period that a neighbor is considered alive after the local receives the last Hello packet from that neighbor. This period is in seconds and is the product of the *hello-inactivity-factor* and the peer-neighbor *hello-interval*.
PTSE lifetime factor	The value for the lifetime multiplier is a percentage. The switch uses it to generate the initial value for the remaining lifetime of a self-created PTSE. This remaining lifetime is the product of the *PTSE lifetime factor* and the *PTSE-refresh-interval*.
Retransmit int	The number of seconds between re-transmissions of unacknowledged DS, PTSE request, and PTSP.
AvCR proportional PM	The proportional multiplier is a percent that used in the algorithms that determine significant change for AvCR parameters.
CDV PM multiplier	The proportional multiplier is a percent that is used in the algorithms that determine significant change for peak-to-peak cell delay variation (CDV).
AvCR minimum threshold	The minimum threshold is a percent that is used in the algorithms that determine significant change for AvCR parameters.
CTD PM multiplier	This proportional multiplier is a percent that is used in the algorithms that determine significant change for cell transfer delay (CTD) parameters.
Peer delayed ack int	The minimum interval between transmissions of delayed PTSE acknowledgment packets appears as 100-millisecond increments. Units: 100 ms.
Logical horizontal link inactivity time	The value of **-horizontalLinkInactivityTime**. The current release does not support this parameter. ??

Related Commands

cnfpnni-timer

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display PNNI timer values. Since the value of *node-index* = 1, the command line displays the PNNI timer values for only node 1.

```
SanJose.7.PXM.a > dsppnni-timer 1
node index: 1
Hello holddown(100ms)... 120 PTSE holddown(100ms)... 120
Hello int(sec).......... 15 PTSE refresh int(sec).. 1800
Hello inactivity factor. 5 PTSE lifetime factor... 200
Retransmit int(sec)..... 5
AvCR proportional PM.... 3 CDV PM multiplier...... 25
AvCR minimum threshold.. 50 CTD PM multiplier...... 50
Peer delayed ack int(100ms)................... 10
Logical horizontal link inactivity time(sec).. 10

Geneva.7.PXM.a >
```

Logical Node, Port, and Signaling Commands

This chapter describes primarily the PNNI commands that apply to PNNI ports. It also contains some node-level commands, such as a variety of commands that relate to node-level congestion thresholds. In general, the commands let you create, delete, display logical elements, and view and clear statistics. The commands pertain to the following areas of PNNI port function:

- Address assignment
- Address filtering
- Address ping
- Signalling
- Control Channels
- Defaults for bandwidth parameters
- Connection-related commands—either or both SVCs and SPVCs
- Port resource commands
- IISP interfaces
- ILMI Port Management
- SSCOP (service-specific connection-oriented protocol)
- Congestion thresholds for the node and for individual ports

Types of Address

Three types of local addresses exist:

- ILMI-registered addresses
- User-provisioned addresses via **addaddr**
- Host application addresses, such as AESA-Ping, PNNI logical group numbers, IP connectivity, node-level SPVC prefix, and so on

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (**–timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 6-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850node uses the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation*, Release 2.

Table 6-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–2	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 6-2 and Table 6-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 6-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 6-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf*.]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical

port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 6-4.

Table 6-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 6-4 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1
- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

Dot Notation in ATM Addresses

In this chapter, many examples of ATM addresses appear with dots. The dots serve only to aid you when entering or viewing these fields, which have considerable lengths. The controller strips out the dots.

addaddr

Add Address—create an ATM address.

Use **addaddr** to specify one or more ATM addresses for a port. The port can be a UNI or an IISP port. For each port, the mandatory parameters are an ATM address and the length of that address. For a description of the format of an ATM address and address planning in general, see the *Cisco MGX 8850 Software Configuration Guide*, Release 2 or higher.

The optional parameters for **addaddr** let you specify:

- Details about the application of the address to either a public or private UNI or an IISP
- An address plan—E.164 or NSAP
- Whether the node at the near end of an IISP link can distribute the new address to the node at the far end of the IISP link (making the new address visible to the nodes in other networks)
- The scope of the ATM address

Note Before you add an ATM address on a UNI port, disable ILMI address registration on the port. To disable ILMI address registration, use **cnfaddrreg** (and supply the portID followed by "no").

Cards on Which This Command Runs

PXM45

Syntax

```
addaddr <portid> <atm-address> <length>
[-type {int | ext}]
[-proto {local | static}]
[-plan {e164 | nsap}]
[-scope value]
[-redst {yes | no}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
atm-address	The ATM address: its format depends on whether the address type is NSAP or E.164. The *address plan* specifies the address type and so determines the *maximum* number of bytes or bits in the address. You can specify the address plan with the forthcoming **-plan** option. The default plan is NSAP. • An NSAP address can have 1–20, 8-bit, hexadecimal bytes. Cisco recommends that you use 20 bytes for the NSAP address. • An E.164 address can 8–15 decimal digits. The number of bits or bytes in the ATM address effects the uniqueness of the address. The longest address ensures total uniqueness of the address. With a one-byte address, any caller that sends an address whose first address byte matches that one-byte ATM address goes to that port.

length	Address length. The units of measure differ for each address plan. The **-plan** option lets you specify E.164 or NSAP. • For an NSAP address plan, the units of measure are bits. The range is 0–160. Using the maximum of a 20-byte ATM address: 20 bytes x 8 bits per byte = 160 bits. • For an E.164 address plan, the value is the number of decimal digits. If the ATM address consists of 15 digits, the value for this parameter is also 15.
-type	The type of reachability of the node. The reachability is either *internal* or *external*. For internal, the address of this port is advertised to only the nodes within the current PNNI network. The default is internal. The external address can go outside the PNNI network and applies to an IISP link or a public UNI. For example, the boundary node on the far side of an IISP link must have access to the ATM address of the near-side boundary node to be able to reach the near-side boundary node. Note that, for any ATM address on an IISP port or a public UNI, you must specify *external*. Possible entries: “internal” (or just “int”” or “external (or just “ext”). Default: int
-proto	The protocol for advertising a reachable address: • If the **-type** is internal, enter **local** for the **-proto** parameter. • If the **-type** is extenal, enter **static** for the **-proto** parameter. Possible entries: local or static. Default: local
-plan	The address plan: E.164 or NSAP. If you choose NSAP address, the first byte of the address implies one of the three NSAP address plans: NSAP E.164, NSAP DCC, or NSAP ICD. For example, 47 is reserved for NSAP ICD (see Example section). Valid entries: e164 or nsap Default: nsap
-scope	The PNNI hierarchal level to which the address is advertised. In a single-peer group (SPG), only ‘0’ applies. This release does not support multiple peer groups. Range: 0–104 Default: 0
-redst	Enable for distribution of a static address. Enter “yes” to enable (distribute) or “no” to disable (do not distribute). Enabling this option means that the address you are now adding is visible to all nodes within the PNNI network. Other networks cannot see specific port addresses unless you enable such addresses for distribution. Default: no

Related Commands

deladdr, **dspaddr**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

For logical port 11:2.8:28, specify the address that follows. Specify its length in bits (160), and leave all optional parameters in the default state. Use **dspaddr** to confirm the address. Note that the ICD code differs from the default from Cisco Systems. The address is:

47.0077.6400.0000.0000.0ca7.9e01.4000.0c81.8000.00

```
Geneva.7.PXM.a > addaddr 11:2.8:28 47.0077.6400.0000.0000.0ca7.9e01.4000.0c81.8000.00 160

Geneva.7.PXM.a > dspaddr 11:2.8:28

47.0077.6400.0000.0000.0ca7.9e01.4000.0c81.8000.00
length: 160   type: internal proto: local
scope: 0 plan: nsap_icd redistribute: false


Geneva.7.PXM.a >
```

addfltset

Add Filter Set—create or modify an ATM address filter set.

A filter controls the access of incoming calls to a port. The **addfltset** command lets you create a new filter set or modify the contents of an existing filter set. Note that, if you want to change the address plan, you must delete the filter set and re-create it. You can assign more than one filter to a port.

Note The **addfltset** command *creates* but does not *associate* a filter set to a port. To associate a filter set to a port, use the **cnf-pnportacc** command.

Cards on Which This Command Runs

PXM45

Syntax

```
addfltset <name>
[-address atm-address <-length address-length> [-plan {nsap | e164}]
[-list {calling | called}]]
[-index number]
[-accessMode {permit | deny}]
[-cgPtyAbsentAction {permit | deny}]
[-cdPtyAbsentAction {permit | deny}]
```

Syntax Description

name	The name of the filter set can have up to 30 characters.
-address	The ATM address. The plan determines the possible number of bytes or bits in the address: • An NSAP address can have 1–20 eight-bit bytes (where a byte is two hexadecimal numbers). A 20-byte address is an *exact* address, and less than 20 bytes is a *prefix*. • An E.164 address can have 8–15 decimal digits. A 15-digit address is an *exact* address, and less than 15 digits is a *prefix*. You can specify the address plan with the forthcoming **-plan** option. NSAP is the default plan. Note that the number of bits or bytes in the ATM address effects the uniqueness of the address. Although the maximum number of characters in the address requires the most key-strokes, the address with the maximum length insures the uniqueness of the port. To use an extreme example: with a one-byte address, for any caller that sends an address whose first address byte matches that one-byte ATM address, the node routes the call to that port. The default is modifying the **accessMode** field of a filter element using the index only, in which case you do not need to specify the address field.

-length	Address length. The units of measure differ for each address plan. The **-plan** option lets you specify E.164 or NSAP. • For an NSAP address plan, the units of measure are bits. The range is 0–160. Using the maximum of a 20-byte ATM address: 20 bytes x 8 bits per byte = 160 bits. For a prefix, you must follow the significant bytes with 3 dots (and no spaces). See Example. • For an E.164 address plan, the value is the number of decimal digits. If the ATM address consists of 15 digits, the value for this parameter is also 15. For a prefix, you must follow the significant digits with 3 dots (and no spaces). See Example section.
-plan	Address plan: **e164** or **nsap**. This option applies only if you specify an address (see **-address**). Default: **nsap**
-list	Address list: “calling” or “called.” You can specify this field only if you also specify the address field. Default: calling
-index	Order in which the filter is applied. If you assign more than one filter to a port, you must plan the order in which the node applies the filters to a calling party. Plan the filters and their order of application so that the order of application does not negate the purpose of filtering. The first position in the filtering order is 1. The **dspfltset** command displays existing filters. Range: 1–65535 Default: 1
-AccessMode	The access mode specifies whether the port permits or denies a call if the address pattern-matching results in a match. Type the entire word “permit” or “deny.” Default: permit
-cgPtyAbsentAction	The access mode specifies whether the port permits or denies a call if the address of the calling party does not match an address in the *calling party* list of the filter. Type the entire word “permit” or “deny.” Default: permit
-cdPtyAbsentAction	The access mode specifies whether the port accepts or denies the call if the calling party does not match any address entry in the *called party* list of the filter. Type the entire word “permit” or “deny.” Default: permit

Related Commands

cnffltset, delfltset, dspfltset, cnf-pnportacc

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

On the port with prefix 47.1111.1111, create address filter “mendocino.” The 47 indicates NSAP ICD address plan. Note that the three-dot notation is necessary for the prefix, and the address length of 40 is the 40 bits that make up that hex prefix.

```
Unknown.1.1.PXM.a > addfltset mendocino -address 47.1111.1111... -length 40 -index 2
```

addpnport

Add Port—adds a UNI or NNI port through PNNI.

The **addpnport** command lets you *pre-configure* an NNI or UNI port. The purpose of pre-configuring is that it may best serve the configuration strategy of some companies.

To pre-configure a port means you add it on the PXM45 before you add it on the service module (by using **addport**). Eventually, you must run **addport** on the service module. The **addpnport** command is optional because when you create the port by using **addport** (and a resource partition by using **addpart**), PNNI automatically creates the port. Therefore, after you create the port on the service module, using the **addpnport** command is not necessary.

After you pre-configure the port through **addpnport** at the controller, its administrative and operational states are *down* by default. Use **uppnport** to bring up the port.

The **addpnport** command only creates the port. The PNNI commands for configuring the operational characteristics of the port are **cnfpnportrange**, **cnfpnportcac**, and **cnfpnportsig**. If the configuration you specify with these commands conflict with the values you specify on the service module through the **addport** or **cnfport** commands, these PNNI commands override the slave-side configuration.

Note that the format of the PNNI logical port and the format of the logical port on the slave side map to each other. See AXSM Format and PNNI Format, page 6-4, for a description of this mapping.

Cards on Which This Command Runs

PXM45

Syntax

addpnport <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

On the PXM45: **delpnport**, **dnpnport**, **uppnport**, **dsppnports**, **dsppnport, cnfpnportcac**, **cnfpnportsig**, **cnfpnportrange**

On the service module: **addport**, **cnfport**, **dspport**, **dspports**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Add a port with the ID 6:1.1:1. Use the **dsppnports** command to see if 6:1.1:1 appears in a list of all ports.(If many PNNI ports already existed, you could use the **dsppnport** command for 6:1.1:1) It appears at the end of the list—after the entries for BITS clock ports in the format 7.x.

Note that the IF status and the Administrative status are both up. (The IF refers to the logical interface on the service module—the slave side rather than the PNNI controller side.) That IF status and the Administrative status are up indicates that the **addpart** and **addport** commands have already been used on the service module. If these commands had not been executed, IF and Admin would be "provisioning" and "down," respectively. For details on the content of the **dsppnports** command, see its description.

```
M8850_NY.7.PXM.a > addpnport 6:1.1:1

M8850_NY.7.PXM.a > dsppnports
Summary of total connections
(p2p=point to point,p2mp=point to multipoint,SpvcD=DAX spvc,SpvcR=Routed spvc)
Type   #Svcc:   #Svpc:   #SpvcD:  #SpvpD:  #SpvcR:  #SpvpR:  #Total:
p2p:   0        0        0        0        0        0        0
p2mp:  0        0        0        0        0        0        0

                         Total=      0/50000

Summary of total configured SPVC endpoints
Type   #SpvcR   #SpvpR   #SpvcD   #SpvpD   Total
p2p:   0         0         2         0         2
p2mp:  0         0         0         0         0
                         Total=2

Per-port status summary

PortId      LogicalId    IF status      Admin status  ILMI state        #Conns

7.35         17251107    up             up            NotApplicable     0

7.36         17251108    up             up            NotApplicable     0

7.37         17251109    up             up            NotApplicable     0

7.38         17251110    up             up            NotApplicable     0

1:2.1:1      16848897    up             up            UpAndNormal       0

2:2.2:1      16914433    provisioning   up            NotApplicable     0

3:1.1:1      16979969    down           up            Disable           0

3:1.2:2      0           provisioning   down          NotApplicable     0

6:1.1:1      17176577    up             up            Disable           0

M8850_NY.7.PXM.a >
```

aesa_ping

ATM End System Address Ping—ping any ATM end station connected to a PNNI network.

The **aesa_ping** command lets you ping an AESA based on an ATM address that you provide as a *destination address*. If you specify only the destination address, the local node merely looks up that address in its routing table. To actually confirm the reachability of another node, specify the optional parameters for setting up a SVC to send and receive keep-alive packets.

Cards on Which This Command Runs

PXM45

Syntax

```
aesa_ping <destination address>
[-setupcall {yes/no} ]
[-qos {ubr | abr | cbr | vbr_rt | vbr_nrt}]
[-pcr {peak cell rate}]
[-scr {sustain cell rate}]
[-trace {yes/no}]
[-timeout {time out in secs}]
[-data {yes/no}]
[-interval {time}]
```

Syntax Description

destination address	Destination address in NSAP format. Example (the dots are optional): `47.009181000000000500ffde873.00500ffde873.01` Default: (none)
-setupcall	Sets up an SVC call as part of the ping. If you do not include the **setupcall** parameter, the system performs only route lookup for the QoS parameters to the destination. Possible values: yes or no Default: no
-qos	Quality of service (QoS) used for SVC ping connection. This parameter applies only if you enable **setupcall**. Possible values for QoS are: ubr, cbr, vbr, vbr-nrt, and vbr-rt Default: ubr
-**pcr**	Peak cell rate of the ping. This parameter only applies if you enable **setupcall**. Range: 1–100 cells per second Default: 10
-**scr**	Sustained cell rate of the ping. This parameter applies only if you enable **setupcall**. Range: 1–50 cells per second Default: 5

-trace	Enable path trace during ping. This parameter only applies if **-setupcall** is enabled. Possible values: yes or no Default: no
-timeout	Connection timeout for the ping. This parameter applies only if **setupcall** is enabled. Range: 5–120 seconds Default: 5 seconds
-data	If you enable **data**, the switch transfers data then prints statistics at the end of the timeout. Possible values: enable or disable Default: disable
-interval	The interval between the call setup of successive transmissions. This parameter applies only if **setupcall** is enabled. Range: 5–120 seconds Default: 5 seconds

Related Commands

dsppingatmaddr

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Ping the ATM end station with the address 47.00918100000000500ffde873.00500ffde873.0.

```
svcpop1.1.PXM.a > aesa_ping 47.00918100000000500ffde873.00500ffde873.01

Ping Got CLI message, index=0

PING:from PNNI - SOURCE ROUTE
DTL    1 :Number of (Node/port)elements    2

DTL 1:NODE 1::56:160:71:0:145::238:238:238:238:

Port 1:263168

DTL 1:NODE 2::56:160:71:0:145::15:253:232:115:

Port 2:0

Port List :no of ports =    1

Port ID    1:263168

svcpop1.1.PXM.a >
```

clrpnconstats

Clear Port Connection Statistics

Clears call statistics for one port or all ports.

Cards on Which This Command Runs

PXM45

Syntax

clrpnconstats [*portid*]

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dsppnconstats

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

First, check the connection statistics on port 4:1.1:11.

```
Geneva.7.PXM.a > dsppnconstats 4:1.1:11

 Call Statistics for 4:1.1:11
 Incoming Call Attempts:     209           Outgoing Call Attempts:       8

 Incoming Call Success:      6             Outgoing Call Success:        0

 Incoming Call Failures:     0             Outgoing Call Failures:       209

 Incoming Filtering Failures:0             Outgoing Filtering Failures : 0

 Incoming Routing Failures:  0             Outgoing Routing Failures :   209

 Incoming CAC Failures:      0             Outgoing CAC Failures :       0

 Incoming Timer Failures:    0             Outgoing Timer Failures :     0

 Incoming Crankback Failures:0             Outgoing Crankback Failures : 0
```

Clear the call statistics on port 4:1.1:11.

```
Geneva.7.PXM.a > clrpnconstats 4:1.1:11
```

Check the results by executing **dsppnconstats**.

```
Geneva.7.PXM.a > dsppnconstats 4:1.1:11

 Call Statistics for 4:1.1:11
 Incoming Call Attempts:     0            Outgoing Call Attempts:       8

 Incoming Call Success:      0            Outgoing Call Success:        0

 Incoming Call Failures:     0            Outgoing Call Failures:       0

 Incoming Filtering Failures:0            Outgoing Filtering Failures : 0

 Incoming Routing Failures:  0            Outgoing Routing Failures :   0

 Incoming CAC Failures:      0            Outgoing CAC Failures :       0

 Incoming Timer Failures:    0            Outgoing Timer Failures :     0

 Incoming Crankback Failures:0            Outgoing Crankback Failures : 0

Geneva.7.PXM.a >
```

clrsigstats

Clear Signaling Statistics

Clears existing signaling statistics for one port or all ports.

Cards on Which This Command Runs

PXM45

Syntax

`clrsigstats` `[`*`portid`*`]`

Syntax Description

portid	The port in the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See PNNI Format, page 6-4.

Related Commands

cnfsigdiag, **delsigdiag**, **dspsigdiag**, **dspsigstats**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear the signaling statistics on port 3:1.1:11. Thereafter, check the results with **dspsigstats**.

```
Geneva.7.PXM.a > clrsigstats 3:1.1:1
Clearing Signaling Statistics for 3:1.1:1

8850_NY.7.PXM.a > dspsigstats 3:1.1:1

 Signaling Statistics for 3:1.1:1
 Message                         Rcv           Xmt
 -------                         ---           ---
 Call Proceeding                   0             0
 Connect                           0             0
 Connect Ack                       0             0
 Setup                             0             0
 Release                           0             0
 Release Complete                  0             0
 Add Party                         0             0
 Add Party Ack                     0             0
 Add Party Rej                     0             0
 Drop Party                        0             0
 Restart                           0             0
 Restart Ack                       0             0
 Status                            0             0
 Status Enquiry                    0             0
 Alerting                          0             0
 Notify                            0             0
 Progress                          0             0

Last Cause/Diag/Crankback
 --------------------------
 Cause                             0
 Diagnostic                        0       0       0       0
 Src Crankback port count          0

8850_NY.7.PXM.a >
```

clrsscopstats

Clear SSCOP Statistics

The **clrsscopstats** command lets you clear the statistics for the service-specific connection-oriented protocol (SSCOP). You can specify the statistics for an individual port or all ports on the switch.

Cards on Which This Command Runs

PXM45

Syntax

`clrsscopstats [ portid ]`

Syntax Description

portid	If you do not specify a port, this command clears all SSCOP statistics on the switch. The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dspsscopstats

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear the SSCOP statistics on port 4:1.1:11.

```
Geneva.7.PXM.a > clrsscopstats 4:1.1:11
```

cnfe164justify

Configure E.164 Justification

Specifies whether the E.164 AESAs with the E.164 AFI are converted to the left or right-justified encoding format. For PNNI to search the address correctly, all nodes in the PNNI network must use the same justification.

Cards on Which This Command Runs

PXM45

Syntax

`cnfe164justify left | right`

Syntax Description

left or **right**	Justification of E164 addresses. Type the entire word "left" or "right." The default is left.

Related Commands

dspsvcparm

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Configure right-hand justification for the E.164 AESAs.

```
Geneva.7.PXM.a > cnfe164justify right
```

cnfenhiisp

Configure Enhanced IISP—enable or disable enhanced IISP features.

The **cnfenhiisp** command enables or disables the enhanced IISP feature on the port. This command applies to only IISP ports. When you change the operational state of enhanced IISP, the change does not affect existing calls.

The items that enhanced IISP include are:

- Generic identifier transport (GIT)
- Virtual path service over the IISP
- Added support or nrt-VBR and rt-VBR
- Transport of frame discard specification

Cards on Which This Command Runs

PXM45

Syntax

`cnfenhiisp <portid> {yes | no}`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
yes | **no**	Enter "yes" to enable enhanced IISP or "no" to disable enhanced IISP. Default = no

Related Commands

dspenhiisp

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Enable enhanced IISP on port 11:2.1:1.

```
Geneva.7.PXM.a > cnfenhiisp 11:2.1:1 yes
```

cnffdonaal5

Configure Nodal Frame Discard

The **cnffdonaal5** command lets you determine whether the switch can use frame discard for AAL5 cells. Use the **dspsigparm** command to see the enable status of frame discard for AAL5 cells.

Cards on Which This Command Runs

PXM45

Syntax

`cnffdonaal5 yes | no`

Syntax Description

yes	Install frame discard on the presence of the AAL5 IE. The default is yes.
no	Do not install frame discard on the presence of the AAL5 IE.

Related Commands

dspsigparm

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Disable frame discard for AAL5 cells on the switch.

```
Geneva.7.PXM.a > cnffdonaal5 no
```

cnffltset

Configure Filter Set—modifies an ATM address filter set.

Use **cnffltset** to modify an existing filter set. This command can:

- Add more addresses to the filter set.
- Change the access mode or address field of a filter set entry.

After a filter is modified for a specific port, associate the filter to that port by using **cnf-pnportacc**.

Cards on Which This Command Runs

PXM45

Syntax

```
cnffltset < name >
[-address atm-address -length address-length [-plan {nsap | e164}]
[-list {calling | called}]]
[-index number]
[-accessMode {permit | deny}]
```

Syntax Description

name	A name for the filter set: the maximum is 30 characters.
address	The 1-40 digit NSAP or 1-15 digit E.164 address. You can add the address to a filter set. The default is modifying the **accessMode** field of a filter element using the index only: in this case, the you need not specify the address field.
length	If the address is specified, the length must be specified. • For NSAP, the length is in bits. • For E164, the length is in bytes.
plan	Address Plan: **e.164** or **nsap**. You may specify this field only if you also specify the address field. Default = **nsap**
list	Address List: **calling** or **called**. You may specify this field only if you also specify the address field. Default = **calling**
index	Order in which filters are set. Range: 1–65535 Default = 1
AccessMode	Specify the access mode (**permit** or **deny**) to whether accept or deny the call on the port if the address pattern-matching succeeds. Default = **permit**

Related Commands

addfltset, delfltset, dspfltset

Attributes

Log: log	State: active	Privilege: SUPER_GP

cnfintfcongth

Configure Interface Congestion Threshold

The **cnfintfcongth** command lets you configure congestion thresholds for a logical port. The thresholds apply to incoming calls and status enquiries. When the upper congestion limit is reached, the port may block incoming calls and adjust the pace of status enquiries.

You must specify at least one keyword.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfintfcongth <portid>
[ -setuphi {setuphival [-unackedstatenqlo {unackedstatenqloval} ]
[ -unackedstatenqhi {unackedstatenqloval} ]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-setuphi	The number of connection set-up messages per second. Above this number, the condition of set-up messages on the interface is congested. Range: 1–260 calls per second Default: 180
-unackedStatEnqLo	The number of status enquires yet to be acknowledged by peer-to-peer interface. Below this value, the congestion condition for status enquiries at the interface level is dropped. Range: 1–500 messages Default: 40
-unackedStatEnqHi	The number of status enquires yet to be acknowledged by peer-to-peer interface. The interface is considered to be congested with status enquiries when this thresholds is reached. Range: 1–500 messages Default: 100

Related Commands

dspintfcongth, dspnodalcongth

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Configure a congestion threshold of 200 for set-up messages on 6:1.1:1. Check the results by using the **dspintfcongth** command.

```
M8850_NY.7.PXM.a > cnfintfcongth 6:1.1:1 -setuphi 200

M8850_NY.7.PXM.a > dspintfcongth 6:1.1:1

Congestion Thresholds for port : 6:1.1:1

Parameter            Value        unit
---------            -----        ----
setuphi              100          cps
unackedStatEnqLo     40           messages
unackedStatEnqHi     200          messages

M8850_NY.7.PXM.a >
```

cnfnodalcongth

Configure Nodal Congestion Thresholds

The **cnfnodalcongth** command lets you configure congestion thresholds at the node level. The thresholds relate to call setup messages, status enquiries, queue levels, and so on. You must specify at least one optional parameter.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfnodalcongth
[-setuphi {setupHiThreshold}]
[-statenqlo {statusEnqLoThreshold}]
[-statenqhi {statusEnqHiThreshold}]
[-connpendlo {connPendingLo}]
[-connpendhi {connPendingHi} ]
[-incompjour {incompleteJournalCallsHi}]
[-vsiqmild {mildCongPerc}]
[-vsiqmedium {mediumCongPerc}]
[-vsiqsevere {severeCongPerc}]
```

Syntax Description

-setuphi	The number of connection setup messages per second above which the node is congested. Range: 1–1000 connection setup messages per second Default: • 500 on a PXM45/B with R7K processor • 180 in all other cases
-statenqlo	The number of status enquiries per second below which the node is not congested. Range: 1–500 messages per second Default: 100
-statenqhi	The number of status enquiries per second above which the node is congested with status enquiries. Range: 1–500 status enquiries per second Default: 200 status enquiries per second
-connpendlo	The aggregate number of connections in the establishment phase below which the establishment congestion flag is dropped. Range: 1–1000 connections Default: 400 connections

-connpendhi	The aggregate number of connections in the establishment phase above which the establishment congestion state is flagged. Range: 1–1000 connections Default: 500 connections
-incompjour	The number of incomplete journaling cycles that must be exceeded to trigger an increase in the journaling rate. Range: 1–10 cycles Default: 5 cycles
-vsiqmild	The VSI Q depth above which VSI master is mildly congested. The *mildCongPerc* value is a percentage of VSI master-slave communication window size. This threshold applies to all the PNNI logical ports on the node. Range: 1–175 Default: 5
-vsiqmedium	The VSI Q depth above which VSI master is congested at a medium level. The *mediumCongPercis* value is a percent of VSI master-slave communication window size.This threshold applies to all interfaces on the node. Range: 1–175 Default: 10
-vsiqsevere	The VSI Q depth above which VSI master is severely congested. The *severeCongPerc* value is a percent of VSI master-slave communication window size. This threshold applies to all interfaces on the node. Range: 1–175 Default: 20

Related Commands

`dspnodalcongth`

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

Configure the nodal congestion thresholds, as follows:

```
svcpop1.1.PXM.a > cnfnodalcongth -setuphi 80 -vsiqmild 100 -vsiqmedium 140 -vsiqsevere 175

svcpop1.1.PXM.a > dspnodalcongth
Parameter         Value      Unit
=================================
setuphi(prov)       80       cps
setuphi(curr)       80       cps
statenqlo          100       cps
statenqhi          200       cps
connpendinglo      400       messages
connpendinghi      500       messages
incompjournalhi      5       cycles
vsiqdepthmild      100       multiplier
vsiqdepthmedium    140       multiplier
vsiqdepthsevere    175       multiplier
```

cnfpnctlvc

Configure Port Control Virtual Channel—configure bandwidth parameters for a control channel.

The **cnfpnctlvc** command lets you configure bandwidth parameters for two types of control channels on a port. The types of control channels are service-specific connection-oriented protocol (SSCOP) and PNNI routing control channel (PNNI-RCC).

Before using **cnfpnctlvc**, note the following:

- You can execute **cnfpnctlvc** for one type of control VC at a time.
- A control VC belongs to a special service type called virtual switch interface signaling (VSI-SIG). A VSI-SIG connection behaves like a VBR-nrt connection.
- The port must be down for you to execute **cnfpnctlvc** (so you may first have to use **dnpnport**).
- The bandwidth used by control-type VCs (including ILMI, when enabled) adds to the bandwidth load on the port. Use **dspload** to determine the load on port resources.

(The VC for ILMI is also a control channel, but its bandwidth parameters are fixed, as follows: PCR=1000 cps; SCR=50cps; and MBS=1024 cells).

The controller guarantees bandwidth for control VCs. The reserved bandwidth is proportional to the line rate (T3/E3, OC3, OC12, and so on). Also, the excess priority for this category is equal to CBR and is higher than any other category. This scheme protects the flow of control data from bursts in user data.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnctlvc <portid> <vc-type>[<-pcr> {peak cell rate}][<-scr> {sustained cell rate}]
[<-mbs> {Maximum burst size}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
vc-type	The type of control VC. The VC types are **pnnircc** and **sscop.**
-pcr	The peak cell rate for the control VC. Range: 1–5000 cells per second Defaults: • If **vc-type** is **pnnircc**, the default is 906 cps. • If **vc-type** is **sscop**, the default depends on the line rate.

-scr The sustained cell rate of the control VC.

Range: 1–5000 cps

Default:

- If **vc-type** is **pnnircc**, the default is 453 cps.
- If **vc-type** is **sscop**, the default depends on the line rate.

-mbs The maximum burst size of the control VC.

Range: 1–1000 cells

Defaults:

- If **vc-type** is **pnnircc**, the default is 171 cells.
- If **vc-type** is **sscop**, the default is 1000 cells.

Related Commands

dsppnctlvc (on the active control card), **dspload** (on a service module)

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Change the MBS of the PNN-RCC VC on port 3:1.1:1 to 100 cells. The system returns the error message that the port is still in service. After downing the port with the **dnpnport** command, complete the task. (If connections were on the port, they would go into alarm.) Check the control VCs with **dsppnctlvc**.

One section of the display (labeled "provisioned") shows *configured* parameters, and one section shows *operational* VC parameters. In this example, the "provisioned" section reflects that you did not configure bandwidth parameters other than MBS, and the "operational" section shows actual bandwidth parameters—including the MBS you configured. Remember to re-activate the port by using **uppnport**.

```
8850_NY.8.PXM.a > cnfpnctlvc 3:1.1:1 sscop -mbs 100

ERROR: Port is not out-of-service

8850_NY.8.PXM.a > dnpnport 3:1.1:1

8850_NY.8.PXM.a > cnfpnctlvc 3:1.1:1 sscop -mbs 100
cnfpnctlvc Successful

8850_NY.7.PXM.a > dsppnctlvc 3:1.1:1

vc type = sscop      Parameter = Provisioned
service category : sig                     PCR : Not Provisioned
SCR :  Not Provisioned                     MBS :             100

vc type = sscop      Parameter = Operational
service category : sig                     PCR :          308000
SCR :            2000                      MBS :             100

8850_NY.8.PXM.a > uppnport 3:1.1:1
```

cnfpnportacc

Configure Port Access—associates an ATM filter set with a port.

The **cnfpnportacc** command lets you *associate* an ATM filter set with a port (after you *create* the filter set with **addfltset**). You must specify at least one filter set if you use this command.

Cards on Which This Command Runs

PXM45

Syntax

`cnfpnportacc <portid> [-in in-filter-name] [-out out-filter-name]`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-in	*in-filter-name*: the name of the filter set that applies to SETUP messages that arrive at the port.
-out	*out-filter-name*: the name of the filter set that applies to SETUP messages that leave the port.

Related Commands

delpnportacc, **addfltset**

Attributes

Log: log	State: active	Privilege: GROUP1

cnfpnportcac

Configure Port Connection Admission Control—configure CAC for a all connections on a port.

The **cnfpnportcac** command lets you reserve a percent of the bandwidth parameters for an individual service type. The new configuration applies to new, incoming calls, while existing calls remain unaffected. You can execute this command whether the port is active or in the provisioning state.

The *bookfactor* is the percent of utilization. It applies to connection admission control (CAC). The booking factor does not apply to the available cell rate (AvCR) advertised by the switch to the controller. When the default for **maxbw** and **minbw** is used for all service types, the common AvCR is advertised for all the service types.

You must specify at least one of the optional keywords.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnportcac <portid> <service_category>
[-bookfactor utilization-factor]
[-maxbw max-bw-percent]
[-minbw min-bw-percent]
[-maxvc max-vc-percent]
[-minvc min-vc-percent]
[-maxvcbw max-vc-bw]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
service_category	Service category (service type). Choices are: cbr, rtvbr, nrtvbr, ubr, or abr.
-bookfactor	The service category utilization factor (SCUF) for a service type. Range: 1–200 Default: 100
-maxbw	The *max-bw-percent*: maximum percentage of bandwidth for a service category on this port. Range: 0–100.0000 Default: 100
-minbw	The *min-bw-percent*: minimum percentage bandwidth for a service category on this port. Range: 0–100.0000 Default: 0

-maxvc	The *max-vc-percent* is the maximum percentage of VCs for a service category on this port. Range: 0–100 Default: 100
-minvc	The *min-vc-percent* is the minimum percentage of VCs for a service category on this port. Range: 0–100 Default: 0
-maxvcbw	The *max-vc-bw* is the maximum number of cells per second specified by the PCR allowed for a VC in a service category on this port. Range: 0 through the maximum possible line rate Default: 0 (disabled)

Related Commands

dsppnportcac

Attributes

Log: log	State: active	Privilege: GROUP1

Usage Guidelines

This section uses three examples to describe the booking factor.

1. For no overbooking or oversubscription, suppose that a user has a 100-Mbit link and the booking factor is 100.

- PNNI advertises 100 Mbits to the network.
- The link on the service module is configured for 100 Mbits.

2. A booking factor less than 100% results in link oversubscription because the bandwidth booked for each connection exceeds the configured bandwidth for the connection. This situation is referred to as overbooking.

   Suppose that, for the same 100-Mbit link, the booking factor is 10.

- PNNI advertises 1000 Mbits (calculated by 100 * 100/10 = 1000)
- The link is configured on the service module for 100 Mbits.

3. Booking factors greater than 100% result in link undersubscription, and the bandwidth booked for a connection exceeds the connection's configured bandwidth. This situation is referred to as underbooking.

   For the same 100-Mbit link, the booking factor is 200.

- PNNI advertises 50 Mbits (calculated as 100 * 100/200 = 50)
- The link is configured on the service module for 100 Mbits.
- The policing bandwidth is based on the configured bandwidth and not the book factor. For a 10-Mbit connection, the policing is 10 Mbits, regardless of the booking factor.

cnfpnportcc

Configure Port Call Control

Sets call control parameters for a port. The possible applications of **cnfpnportcc** are to let you:

- Specify maximum root and leaf connections for point-to-multipoint connections
- Specify whether all SVCs or all SPVCs are blocked on the port

Note The current release does not support point-to-multipoint calls.

You can execute this command whether the port is active or in the provisioning state. Changes apply to new incoming calls, not existing calls. You must specify at least one optional keyword.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnportcc <portid>
[-maxp2mproot max-p2mp-root]
[-maxp2mpleaf max-p2mp-leaf]
[-svcblock {yes | no}]
[-spvcblock {yes | no}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-*maxp2mproot*	The *max-p2mp-root*: maximum number of root VCs on this port. Default: 1000
-*maxp2mpleaf*	The *max-p2mp-leaf*: maximum number of leaf VCs on this port. Default: 4095
-svcblock	Enables or disables SVC *blocking* on the port. Yes (enable): The port accepts no call setups. No (disable): The port accepts call setups. Default: no.
-spvcblock	Enables or disables SPVC *blocking* on the port. Type “yes” or “no.” Yes (enable): an attempt to add an SPVC through **addcon** or Cisco WAN Manager fails. The resulting error message is: “ERR: SPVC blocking is enabled on this interface.” No (disable): you can add SPVCs through **addcon** or Cisco WAN Manager. Default: no

Related Commands

dsppnportcc

Attributes

Log: log	State: active	Privilege: GROUP1

cnfpnportloscallrel

Configure PNNI Port Loss of Signal Call Release

The **cnfpnportloscallrel** command lets you shut off the standard delay for rerouting calls on a port when the system detects loss of signal (LOS) on a port.

When the system detects LOS on an NNI link, the switch does not immediately tear down the calls on the link—in case the break is momentary. By default, the system waits for the SSCOP "no-response" and T309 timers to time out before it releases calls on the broken link. The default values for these timers are 30 seconds and 10 seconds, respectively. The system-level assumption (and therefore the default for **cnfpnportloscallrel**) is to retain all the calls for a temporary loss of connectivity, but this can also have the effect of delaying the rerouting of connections. The **cnfpnportloscallrel** command lets you direct the system to reroute calls without delay on a particular port.

Cards on Which This Command Runs

PXM45

Syntax

cnfpnportloscallrel <*portid*> <**yes** | **no**>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
yes | no	Specifies whether immediate call release is enabled upon LOS. To enable this feature—to remove the standard reroute delay—type "yes." Default: no.

Related Commands

dsppnportloscallrel

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Enable call release upon LOS for port 3:1.1:1, then confirm its status.

```
8850_NY.8.PXM.a > cnfpnportloscallrel 3:1.1:1 yes
8850_NY.8.PXM.a > dsppnportloscallrel 3:1.1:1
Call release on Los :enabled
```

cnfpnportrange

Configure Port Range—configure a range of VPIs or VCIs for SVCCs and SVPCs.

The **cnfpnportrange** command lets you specify a range of VPIs and VCIs on a port for SVCCs or SVPCs for the purpose of screening calls from another switch. If the VPI or VCI for one of these control channels from another switch does not fall within the range for the called port, the called switch rejects the call.

To execute this command, the port must be down (see **dnpnport)**. The **maxsvccvpi** default of 4095 provides the maximum advantage for negotiation during ILMI auto-configuration. Note that the **maxsvccvpi** parameter is limited by the highest VPI that the switch assigns (whether the port is a UNI or an NNI).

In addition to the port ID, you must specify at least one keyword for **cnfpnportrange**.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnportrange <portid>
[-minsvccvpi min-svcc-vpi ]
[-maxsvccvpi max-svcc-vpi ]
[-minsvccvci min-svcc-vci ]
[-maxsvccvci max-svcc-vci ]
[-minsvpcvpi min-svpc-vpi ]
[-maxsvpcvpi max-svpc-vpi ]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-minsvccvpi	The *min-svcc-vp:* minimum VPI for SVCC. Range: 0–4095 Default: 0
-maxsvccvpi	The *max-svcc-vp:* maximum VPI for SVCC. Range: 0–4095 Default: 4095
-minsvccvci	The *min-svcc-vci:* minimum VCI for SVCC. Range: 0–65535 Default: 35
-maxsvccvci	The *max-svcc-vci:* maximum VCI for SVCC. Range: 32–65535 Default: 65535

-**minsvpcvpi**	The *min-svpc-vp:* minimum VPI for SVPC. Range: 1–4095 Default: 1
-**maxsvpcvpi**	The *max-svpc-vp:* maximum VPI for SVPC. Range: 1–4095 Default: 4095

Related Commands

dsppnportrange

Attributes

Log: log	State: active	Privilege: GROUP1

cnfpnportsig

Configure Port Signaling—specify ATM signaling parameters on a PNNI port.

Before you can execute **cnfpnportsig**, the port must be down (by **dnpnport)**. For a UNI or an NNI, the type of interface you specify on the controller must match the interface type that you configure on the slave (the service module). On the service module, you need specify UNI, NNI, or VNNI, but the **cnfpnportsig** command has options for specifying the version of an interface type and other parameters that apply to signaling. See the Syntax Description.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfpnportsig <portid>
[-univer {uni30 | uni31 none}]
[-nniver {iisp30 | iisp31 | pnni10 | enni}]
[-unitype {public | private}]
[-addrplan {both | aesa | e164}]
[-side {user | network}]
[-vpi <vpi>]
[-sigvci signalling-vci]
[-rccvci routing-vci]
[-cntlvc {ip}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-univer	The UNI version: **uni30**, **uni31**, **none**. Note that *univer* and *nniver* are mutually exclusive—so the interface at each end of the connection must have the same interface type. Also, the port type on the PNNI controller must be the same as on the slave (through **addport** ... *ifType* on the AXSM, for example). The default for this parameter is UNI 3.1. If this version is sufficient, you can forego this parameter. However, to change a UNI version, the port must be down. Remember to up the port by using the **uppnport** command after completing the **cnfpnportsig** command. The **none** choice applies to any port that does not need to run SSCOP protocol (for example, SPVC endpoints). Default: **uni31**

-nniver	The NNI version: **iisp30**, **iisp31**, **pnni10**, or **enni**. Note that *univer* and *nniver* are mutually exclusive—so the interface at each end of the connection must have the same interface type. Also, the port type on the PNNI controller must be the same as on the slave (through **addport** ... *ifType* on the AXSM, for example). The default for this parameter is PNNI 1.0. If this version is sufficient, you can forego this parameter. However, to change an NNI version, the port must be down. Remember to up the port by using the **uppnport** command after completing the **cnfpnportsig** command. Default: **pnni10**
-unitype	The type of UNI is either private or public. This parameter is relevant only if you specified a UNI interface through the **-univer** parameter. Default: **private**
-addrplan	The address plan of the calling party that the interface accepts. The choices are **both**, **e164**, and **aesa**. The default is **both**. Only a public UNI can use this parameter. For all other interface types, the port automatically accepts either AESA or E.164 address plans.
-**side**	The side of the port: type "user" or "network." This parameter applies to IISP only and public UNI. (An NNI interface type automatically is "network.") An IISP or public UNI has a user side and a network side. (If both sides are the same, a configuration error has occurred.) The network side is the side that assigns the VPI and VCI. These links do not distinguish between SVCs and SPVCs. Default: **network**
-**vpi**	The VPI of the signaling and routing control channel (RCC) on the port. Range: 0–4095 Default: 0
-**sigvci**	The signaling VCI for the port. If you do not use the default of 5, this VCI must be in the range 32–65535. Range: 5 or 32–65535 Default: 5
-**rccvci**	The *routing control channel-vci*: the VCI for PNNI RCC. If you do not use the default of 18, this VCI must be in the range 32–65535. Range: 18 or 32–65535 Default: 18
-**cntlvc**	Enable for an IP-based signaling channel. This option applies only to a feeder connected to the switch. An IP-based control channel is mutually exclusive of either UNI or NNI. The only choice for -**cntlvc** is "ip." Default: "ip" (for Internet Protocol)

Related Commands

dsppnportsig

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Specify an RCC VCI of 10000 for port 1:2.1:1. Be sure the interface type is compatible with the parameter you want to change. Do the following:

1. Confirm that the interface type for 1:2.1:1 is NNI.
2. Down the port.
3. Configure the RCC VCI to be 10000.
4. Check the port by using the **dsppnportsig** command.
5. Up the port.

```
M8850_NY.7.PXM.a > dsppnport 1:2.1:1

Port:               1:2.1:1           Logical ID:       16848897
IF status:          up                Admin Status:     up
UCSM:               enable
Auto-config:        enable            Addrs-reg:        enable
IF-side:            network           IF-type:          nni
UniType:            private           Version:          pnni10
Input filter:       0                 Output filter:    0
minSvccVpi:         0                 maxSvccVpi:       200
minSvccVci:         35                maxSvccVci:       255
minSvpcVpi:         1                 maxSvpcVpi:       200

       #SpvcCfg:  #SpvcActive:  #SpvpCfg:  #SpvpActive:
p2p :  0          0             0          0
p2mp:  0          0             0          0
       #Svcc:     #Svpc:        Total:
p2p :  0          0             0
p2mp:  0          0             0
                                Total:     0

M8850_NY.7.PXM.a > dnpnport 1:2.1:1

M8850_NY.7.PXM.a > cnfpnportsig 1:2.1:1 -rccvci 10000

M8850_NY.7.PXM.a > dsppnportsig 1:2.1:1

provisioned IF-type: nni   version:     pnni10
sigType:  private          side:        network
addrPlan:  aesa

sigVpi:          0         sigVci:             5
rccVpi:          0         rccVci:         10000

M8850_NY.7.PXM.a > uppnport 1:2.1:1
```

cnfsig

Configure Signaling—configures signaling timers and crankback attempts for a port.

The **cnfsig** command lets you configure signaling timers for a port whether the port is up or down. The new configuration applies to new incoming calls while existing calls remain intact. In addition to standard timers, this command lets specify the maximum number of crankbacks that PNNI can attempt at a port. You must specify at least one of the optional parameters.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfsig <portid>
[-t301 t301-timer ]
[-t303 t303-timer ]
[-t308 t308-timer ]
[-t310 t310-timer ]
[-t316 t316-timer ]
[-t317 t317-timer ]
[-t322 t322-timer ]
[-t397 t397-timer ]
[-t398 t398-timer ]
[-t399 t399-timer ]
[-maxcrbk value ]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-t301	The T301 timer. Range: 150–240 seconds Default: 180
-t303	The T303 timer. Range: 4–8 seconds Default: 4
-t308	The T308 timer. Range: 20–45 seconds Default: 30

-t310	The T310 timer. Range: • 10–20 seconds for UNI. • 30–120 seconds for PNNI. The range you can *specify* for PNNI is 30–120. If you do not specify a T310 timer value for PNNI, it remains the default of 10 seconds for PNNI. Default: 10
-t316	The *t316-timer*: Set the T316 timer. Range: 90–300 seconds Default: 90
-t317	The *t317-timer*: Set the T317 timer. Range: 60–300 seconds Default: 60
-t322	The *t322-timer*: Set the T322 timer. 4–20 seconds. Default: 4
-t397	The *t397-timer*: Set the T397 timer. Range: 180–240 seconds Default: 180
-t**398**	The *t398-timer*: Set the T398 timer. Range: 4–20 seconds Default: 4
-t399	The *t399-timer*: Set the T399 timer. Range: 14–28 seconds for UNI 3.0 and 3.1. Default: 14.
-maxcrbk	The maximum number of crankback attempts allowed on the port. Range: 0–10 Default: 3

Related Commands

dspsig

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Configure port 1:1.1:1 to have the maximum crankback count of 5. Check the results with the **dspsig** command. Note the default values in the **dspsig** output.

```
pop20one.7.PXM.a > cnfsig 1:1.1:1 -maxcrbk 5

pop20one.7.PXM.a > dspsig 1:1.1:1
Signaling Timers for port: 1:1.1:1

Timer           Value(secs)
-----           -----------
t301            180
t303            4
t308            30
t310            10
t316            90
t317            60
t322            4
t397            180
t398            4
t399            14

Max Crankback: 5
```

cnfsigdiag

Configure Signaling Diagnostic

The **cnfsigdiag** command lets you create a filter table for signaling diagnostics or disable signaling diagnostics. Signaling diagnostics are disabled by default.

The ATM signaling diagnostics are tools for troubleshooting call failures in the network and should not be enabled while the switch is operating.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfsigdiag {[enable | disable | index]}
[-cldaddr nsap-address]
[-clgaddr nsap-address]
[-cldaddrmask atm-address-mask]
[-clgaddrmask atm-address-mask]
[-casttype {all | p2p | p2mp}]
[-clrcause clear-cause-code]
[-connctgy {all | svc | svp | swvc | swvp}]
[-inport portid]
[-outport portid]
[-maxrec max-num-records ]
[-scope {all | ext | int}]
[-servctgy {all | cbr | rtvbr | nrtvbr | ubr | abr}]
[-status {active | inactive}]
```

Syntax Description

enable, **disable**, **or** *index*	Enable or disable signaling diagnostics or configure an index. Specify the diagnostics index number for the filter table and enter the diagnostics configuration mode. The range for *index* is 1–50. If you do not specify an index, the enable or disable condition globally applies to all signaling diagnostics. Default: disable
-cldaddr	The *nsap-address* is the filter for ATM signaling call failures against this called address. Default: NULL
-clgaddr	The *nsap-address* is the filter for ATM signaling call failures against this calling address. Default: NULL

-cldaddrmask	The the *atm-address-mask*: Address mask for identifying valid bits of the called NSAP address field (ff.ff.ff, for example). To match this selection criterion, a failed connect setup must have a called party address value equal to the configured called party address for all bits that are 1 in the specified mask. Default: NULL. NULL means the rejected call matches the filter criteria for any called address in the rejected call.
-clgaddrmask	The *atm-address-mask*: Address mask for identifying valid bits of the calling NSAP address field. (ff.ff.ff, for example). To match this selection criteria, a failed connect setup must have a calling party address value equal to the configured calling party address for all bits that are 1 in the specified mask. Default: NULL means the call matches the filter criteria for any calling address in the rejected call.
-casttype	Filtering by connection type. The types are point-to-point (**p2p**), point-to-multipoint (**p2mp**)—currently not supported, or both (**all**)). Default: all
-clrcause	The *clear-cause-code*: Filters ATM signaling call failures by the release cause code (a decimal number) as specified in the ATM Forum UNI 3.1 specification. Default: 0, meaning the cause code is not considered during filtering.
-connctgy	Filters ATM signaling call failures by virtual circuit category (SPVC, SPVP, SVC, SVP, or all of these circuit categories). Default: all
-inport	The *portid*: filters ATM signaling call failures based on the incoming port of the call. Default: 0, meaning the incoming port is not considered during filtering.
-outport	The *portid*: filters ATM signaling call failures based on the outgoing port of the call. Default: 0, meaning the outgoing port is not considered during filtering.
-maxrec	The *max-num-records*: the maximum number of records collected for a particular signaling diagnostics filter table entry. When the maximum value is reached, the older records are deleted. If this field is set to -1, the records are not overwritten. Setting this field to -1 increases memory usage for call failure records and can lead to shortages of available system memory. Range: –1 through 214783647 Default: 20
-scope	The filtering scope choices are within the switch (**int**), on other switches (**ext**), or both (**all**). Default: **all**

-servctgy	Filters ATM signaling call failures by service category (service type): valid entries are: all (for all service types), cbr, rtvbr, nrtvbr, ubr, or abr. Default: **all**
-status	The status of the entry for the signaling diagnostics filter table. Type **active** to begin filtering failed connections or **inactive** to stop filtering failed connections. The **inactive** specification causes the node to delete all the records associated with the filter entry. Default: **inactive**

Related Commands

delsigdiag, dspsigdiag, dspsigstats, clrsigstats

Attributes

Log: log	State: active	Privilege: ANYUSER

cnfspvcprfx

Configure SPVC Prefix—configure a node-level prefix for SPVCs.

For the node to support SPVCs, it must have a 13-byte SPVC prefix that applies to the entire node. No SPVCs can exist on the node until it has an SPVC prefix. Likewise, to change this prefix, no SPVCs can exist on the node.

Prerequisites

Setting up a node and a network requires advance planning for the PNNI node addressing scheme. For basic guidance on the topic of address planning, refer to related material in the *Cisco MGX 8850 Switch Software Configuration Guide,* Release 2.

Cisco provides a default SPVC prefix that is the same as the Cisco-supplied ATM address prefix. Each of these default prefixes contains an International Code Designator (ICD) that is unique to Cisco Systems. Therefore, Cisco Systems recommends that you change the ICD identifier for both the ATM address prefix and the SPVC prefix if the node is planned for operation in a public ATM network. If the node operates in a private ATM network, it can keep the default ATM and SPVC address prefixes.

The following list shows the order of prerequisite commands and the **cnfspvcprfx** command. All commands run on the PXM45.

1. **addcontroller**
2. **cnfpnni-intf**
3. **cnfspvcprfx**
4. **dspspvcprfx**

Cards on Which This Command Runs

PXM45

Syntax

cnfspvcprfx -prfx <*prefix* | "default">

Syntax Description

-prfx	The prefix is either a 13-byte value you enter or the character string "default" (to keep the Cisco factory default).

Related Commands

dspspvcprfx

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

First display the current SPVC prefix. The ICD field shows the prefix is the default from Cisco (0091). Configure the SPVC prefix 47.0077780000000aa2109ff214.

```
pop20one.7.PXM.a > dspspvcprfx
SPVC Node Prefix: 47.00918100000000107b65f33c

pop20one.7.PXM.a > cnfspvcprfx 47.0077780000000aa2109ff214
```

cnfsscop

Configure SSCOP

The **cnfsscop** command lets you configure service-specific connection-oriented protocol (SSCOP) on a port. You can use this command regardless of the state of the port. You must specify at least one of the optional parameters.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfsscop <portid>
[-polltmr {poll-timer | 0}]
[-keepalivetmr {keepalive-timer | 0}]
[-idletmr {idle-timer | 0}]
[-cctmr {cc-timer | 0}]
[-norsptmr {noresponse-timer | 0}]
[-t309tmr {t309-timer | 0}]
[-maxcc {retries | 0}]
[-sndwnd {send-window-packets | 0}]
[-rcvwnd {recv-window-packets | 0}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-polltmr	Number of seconds to send POLL PDUs in the active phase. A 0 forces restore of the default value. Range: 1–5 seconds Default: 1 second
-keepalivetmr	Number of seconds to send POLL PDUs in the transient phase. **0** forces restore of the default value. Range: 1–10 seconds Default: 5 seconds
-idletmr	Number of seconds to send POLL PDUs in the idle phase. **0** forces restore of the default value. Range: 1–20 seconds Default: 10 seconds
-cctmr	Number of seconds to send BGN/END/RS/ER PDUs at the connection control phase. **0** forces restore of the default value. Range: 1–5 seconds Default: 1 second

-norsptmr	Number of seconds after which at least one STAT PDU must be received (for the No Response timer). A **0** forces restoration of the default value. Range: 1–45 seconds Default: 30 seconds
-t309tmr	Number of seconds before SAAL reconnects after disconnection. Range: 1 –15 seconds Default: 10 seconds
-maxcc	Maximum number of retries for connection control operations. A **0** forces restoration of the default. Range: 1–15 Default: 10
-sndwnd	Number of packets the port can send before it must receive an acknowledgment from the ATM switch. A **0** forces restoration of the default. Range: 1–127 Default: 30
-rcvwnd	Number of packets the port can receive before it sends an acknowledgment to the ATM switch. A **0** forces restoration of the default. Range: 1–127 Default: 30

Related Commands

disablesscop, **dspsscop**, **dspsscopstats**

Attributes

Log: log	State: active	Privilege: GROUP1

cnftrapip

Configure Trap IP

The **cnftrapip** command lets you configure the trap IP for Cisco WAN Manager. You can then use the command **dsptrapip** to confirm the value.

Before you use **cnftrapip:**

1. The SNMP agent must be installed.
2. The switch's interface must have an IP address. (To assign an IP address to a switch's interface, use **ipifconfig**.)

For information about installing the SNMP agent for CWM, see *Cisco WAN Manager Installation for Solaris, Release 10.*

Cards on Which This Command Runs

PXM45

Syntax

cnftrapip <*ip address*>

Syntax Description

ip address	The switch's ethernet IP address on which the traps are configured.

Related Commands

dsptrapip, **dsptrapmgr**, **addtrapmgr**, **deltrapmgr**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Assign IP address 172.27.27.184 to the switch, then use the **dsptrapip** command to check it.

```
SanJose.7.PXM.a > cnftrapip 172.27.27.184

SanJose.7.PXM.a > dsptrapip
Trap IP Address :172.27.27.184

SanJose.7.PXM.a >
```

deladdr

Delete Address

Removes an ATM address for a UNI or IISP.

Cards on Which This Command Runs

PXM45

Syntax

```
deladdr <portid> <atm-address> <length>
[-plan {e164 | nsap}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
atm-address	The ATM address: its format depends on whether the address type is NSAP or E.164. The *address plan* specifies the address type and so determines the *maximum* number of bytes or bits in the address. You can specify the address plan with the forthcoming -**plan** option. The default plan is NSAP. • An NSAP address can have 1–20, 8-bit bytes (where a byte is 2 hexadecimal characters). Cisco recommends that you use 20 bytes for the NSAP address. • An E.164 address can 8–15 decimal digits. The number of bits or bytes in the ATM address effects the uniqueness of the address. The longest address ensures total uniqueness of the address. With a one-byte address, any caller that sends an address whose first address byte matches that one-byte ATM address goes to that port.
length	Address length. The units of measure differ for each address plan. The **-plan** option lets you specify E.164 or NSAP. • For an NSAP address plan, the units of measure are bits. The range is 0–160. Using the maximum of a 20-byte address: 20 bytes x 8 bits per byte = 160 bits. • For an E.164 address plan, the value is the number of decimal digits. If the ATM address consists of 15 digits, the value for this parameter is also 15.
-plan	Address plan: E.164 or NSAP. Default: nsap

Related Commands

addaddr, dspaddr

Attributes

Log: log	State: active	Privilege: GROUP1

Example

On port 11:2.8:22, delete 47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8000.00. Note that the command entry includes the address length of 160 after the address.

```
Geneva.7.PXM.a > deladdr 11:2.8:28 47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8000.00 160
```

delfltset

Delete Filter Set

Removes an ATM address filter set.

Cards on Which This Command Runs

PXM45

Syntax

```
delfltset <name>
[index]
```

Syntax Description

name	The name of the filter set.
index	Delete a specific filter element. Range: 1–65535 Default: 0

Related Commands

addfltset, cnffltset, dspfltset

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Remove the “connect” filter set from the port.

```
Geneva.7.PXM.a > delfltset connect
```

delpnport

Delete Port

Remove a UNI or NNI port from the controller. It is allowed only if the PNNI port does not exist on the switch i.e, the PNNI partition associated with the port is removed. If the controller does no provisioning on the port in the Plug and Play Scenario, then the PNNI port on the controller is removed if you remove the PNNI Partition on the switch.

Cards on Which This Command Runs

PXM45

Syntax

delpnport <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

addpnport, **uppnport**, **dnpnport, dsppnport**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Delete port 11:2.8:28 from the switch.

```
Geneva.7.PXM.a > delpnport 11:2.8:28
```

delpnportacc

Delete Port Access

Removes an address access filter group from a port.

Cards on Which This Command Runs

PXM45

Syntax

```
delpnportacc <portid>
{in | out}
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
in	Delete incoming access group for the port.
out	Delete outgoing access group for the port.

Related Commands

cnfpnportacc

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Delete the filter group for incoming calls from port 11:2.8:28.

```
Geneva.7.PXM.a > delpnportacc 11:2.8:28 in
```

delsigdiag

Delete Signaling Diagnostic

Removes a signaling diagnostics filter table entry or removes some configuration values within a filter table entry.

Cards on Which This Command Runs

PXM45

Syntax

```
delsigdiag [index]
[-cldaddr nsap-address]
[-clgaddr nsap-address]
[-cldaddrmask {yes | no}]
[-clgaddrmask {yes | no}]
[-casttype {yes | no}]
[-clrcause {yes | no}]
[-connctgy {yes | no}]
[-inport {yes | no}]
[-outport {yes | no}]
[-maxrec {yes | no}]
[-scope {yes | no}]
[-servctgy {yes | no}]
```

Syntax Description

index	Specifies the diagnostics index number for the filter table. If no other keywords are entered, the indexed filter table entry is deleted. Range: 1–50
-cldaddr	Removes the configured called address from the filter entry.
-clgaddr	Removes the configured calling address from the filter entry.
-cldaddrmask	**yes** returns the called address mask to the default. Default: **no**
-clgaddrmask	**yes** returns the calling address mask to the default. Default: **no**
-casttype	**yes** to disable filtering by connection type. Default: **no**
-clrcause	**yes** to disable filtering by the release cause code. Default: **no**
-connctgy	**yes** returns the connection category to the default. Default: **no**

-inport	**yes** returns the incoming port to the default. Default: **no**
-outport	**yes** returns the outgoing port to the default. Default: **no**
-maxrec	**yes** returns the maximum records to the default. Default: **no**
-scope	**yes** to disable filtering by scope. Default: **no**
-servctgy	**yes** returns the service category to the default. Default: **no**

Related Commands

cnfsigdiag, delsigdiag, dspsigstats, clrsigstats

Attributes

Log: log	State: active	Privilege: ANYUSER

disablesscop

Disable SSCOP

The **disablesscop** command lets you disable SSCOP on a port. The port must be administratively down (by the **dnpnport** command). Be sure a valid and useful reason exists for disabling SSCOP.

Cards on Which This Command Runs

PXM45

Syntax

disablesscop <*portid*> {**yes** | **no**}

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
yes *or* **no**	The choice for disabling or enabling SSCOP on the port: • yes means disable SSCOP. • no means enable SSCOP. Default: no (enable SSCOP on the specified port)

Related Commands

cnfsscop, dspsscop, dspsscopstats

Attributes

Log: nolog	State: active	Privilege: GROUP1

Example

Disable SSCOP on port 11:2.8:28.

```
Geneva.7.PXM.a > disablesscop 11:2.8:28 yes

Geneva.7.PXM.a >
```

dnpnport

Down PNNI Port

The **dnpnport** command takes a UNI or NNI port out of service (administratively "downs" a port). For example, downing a port is necessary for certain provisioning activity or maintenance activity. Where appropriate, the applicable commands state that you must down a port by using **dnpnport**.

Note This command deletes all connections on a port—except for SPVCs whose endpoints are on the port.

Cards on Which This Command Runs

PXM45

Syntax

`dnpnport` <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

addpnport, **delpnport**, **uppnport**, **dsppnport**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Remove port 11:2.8:22 from service.

```
Geneva.7.PXM.a > dnpnport 11:2.8:28

Geneva.7.PXM.a >
```

dspaddr

Display Address

The **dspaddr** command displays ATM addresses and associated information for a UNI or IISP. For details on the displayed items, see the **addaddr** description. The items in the display are as follows:

- ATM addresses on the port and the length of each
- Address plan
- The type of address—internal or external
- The protocol for advertising the address
- The choice for address distribution
- The scope—applicable to multi-peer groups only

Cards on Which This Command Runs

PXM45

Syntax

dspaddr <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

addaddr, deladdr

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the ATM address and other details for port 3:1.1:1.

```
8850_NY.8.PXM.a > dspaddr 3:1.1:1
47.0091.8100.0002.0003.6b5e.30c0
length: 104    type: internal      proto: local
scope: 0       plan: nsap_icd      redistribute: false

8850_NY.8.PXM.a >
```

dspatmaddr

Display ATM Addresses—displays active ATM addresses on a port.

The output of **dspatmaddr** includes configured ATM addresses, configured ILMI address prefixes, and ATM addresses registered by way of the ILMI address registration procedure.

Cards on Which This Command Runs

PXM45

Syntax

dspatmaddr <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dspaddr

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Examples

Display the ATM addresses and prefixes on port 2:1.1:1.

```
Geneva.7.PXM.a > dspatmaddr 2:1.1:1
Port Id: 2.1
Configured Port Address(es):
    39.840f.8011.3744.0000.0040.1005.3456.7834.7777.77
    length: 160           type: internal      proto: local
    scope:  0             plan: nsap-dcc      redistribute: false

ILMI Configured Port Prefix(es):
    47.0091.8100.0000.0000.0ca7.9e01
    88.8888.8888.0000.0000.0000.0000

ILMI Registered Port Address(es):
    47.0091.8100.0000.0000.0ca7.9e01.1234.5678.9012.34
    88.8888.8888.0000.0000.0000.0000.1234.5678.9012.34

length: 160      type: internal      proto: local
scope:  0        plan: nsap-dcc      redistribute: false

Geneva.7.PXM.a >
```

dspconsegep

Display Connection Segment Endpoint

Displays oam segment endpoint for a connection endpoint. When both vpi and vci are present, the segment endpoint is a F5 flow endpoint (for VCCs). When the optional vci is not present, the segment endpoint is a F4 flow endpoint (for VPCs). This command applies to established calls only.

Cards on Which This Command Runs

PXM45

Syntax

```
dspconsegep <portid> vpi [vci]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
vpi	The VPI of the connection.
vci	The VCI of the connection.

Related Commands

cnfoamsegep, dspoamsegep, cnfconsegep, delconsegep

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

dspenhiisp

Display Enhanced IISP

Displays the enhanced IISP setting for one port or all ports.

Cards on Which This Command Runs

PXM45

Syntax

dspenhiisp [<*portid*>]

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfenhiisp

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
Geneva.7.PXM.a > dspenhiisp
Enhanced IISP Features Setting
 Port Id     Enabled
 <portid1>   yes
 <portid2>   no
 ...
Geneva.7.PXM.a >
```

dspfltset

Display Filter Set

Displays a specific ATM filter set or a summary of ATM filter sets.

Cards on Which This Command Runs

PXM45

Syntax

```
dspfltset [-name <...>]
```

Syntax Description

-name	If you do not type a filter set name, a summary of ATM filter sets is displayed.

Related Commands

addfltset, cnffltset, delfltset

Attributes

Log: log	State: any_state	Privilege: ANYUSER

Example

```
Geneva.7.PXM.a > dspfltset SanJose
FilterName: SanJose
Index: 1
Address: 1234567890123456789012345678901234567890
AddrLen: 160 bits
AddrPlan: Nsap
AccessMode: Permit
AddrList: Calling
---------------------------------------
FilterName: SanJose
Index: 2
Address: 1234567890123456789012345678901234567891
AddrLen: 160 bits
AddrPlan: Nsap
AccessMode: Deny
AddrList: Called
---------------------------------------
Output example for the command
dspfltset
Filter Number: 1
FilterName: SanJose
ScreeningDigits: 40
CgPtyAbsentAction: Permit
CdPtyAbsentAction: Deny
---------------------------------------
Filter Number: 2
FilterName: Sunnyvale
ScreeningDigits: 20
CgPtyAbsentAction: Deny
CdPtyAbsentAction: Deny

Geneva.7.PXM.a >
```

dspintfcongcntr

Display Interface Congestion Counters

Display the congestion thresholds for a port. This command cannot be used if the interface is in PROVISIONING state or if the interface (“IF” in the display) is down.

Cards on Which This Command Runs

PXM45

Syntax

dspintfcongcntr <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dspintfcongth, **cnfintfcongth**

Attributes

Log: nolog	State: active, standby	Privilege: SUPER_GP

Examples

```
Geneva.7.PXM.a > dspintfcongcntr 11:2.2:22
Parameter            Value          thresh1 thresh2
============================================================
setupRx                  0             140     180
unackstatenq             0              40     100

Parameter            Value            Mild     Medium   Severe
============================================================
vsiqdepth                0               5      10       20

Geneva.7.PXM.a >
```

Note: *vsiqdepth* is the depth of the VSI queue for the slave that serves the interface identified by *portid*, and this is displayed as the percentage of VSI communication window size.

dspintfcongflags

Display Interface Congestion Flags

To display various congestion detection and action flags maintained at the CCM at the interface level.

Cards on Which This Command Runs

PXM45

Syntax

dspintfcongflags <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dspnodalcongflag

Attributes

Log: nolog	State: active, standby	Privilege: SUPER_GP

Examples

```
Geneva.7.PXM.a > dspitfcongflags 4:1.1:1

Parameter                Value
===================================
vsimildcongflg           FALSE
vsimedcongflg            FALSE
vsiseverecongflg         FALSE
setupflg                 FALSE
unackstatenqcongflg      FALSE
Congestion Action Flags for Interface:1.4

Parameter                Value
===================================
dropsetupflg             FALSE
dropestabflg             FALSE
queuerel                 FALSE
markcallsforrelflag      FALSE
pacevsiresyncflg         FALSE
pacestatenqflg           FALSE
speedjournalflg          FALSE
pacepnniroutecalflg      FALSE
lowersetupthflg          FALSE
```

dspintfcongth

Display Interface Congestion Thresholds

The **dspintfcongth** command display the congestion thresholds for a port.

Cards on Which This Command Runs

PXM45

Syntax

dspintfcongth <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfintfcongth

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

```
svcpop1.1.PXM.a > dspintfcongth 10.1

 Congestion Thresholds for port : 10.1

 Parameter            Value        unit
 ---------            -----        ----
 setuphi              90           cps
 unackedStatEnqLo     40           messages
 unackedStatEnqHi     100          messages

Geneva.7.PXM.a >
```

dspnodalcongcntr

Display Nodal Congestion Threshold Counters

The **dspnodalcongcntr** command displays thresholds and current contents of the congestion counters for the node.

Cards on Which This Command Runs

PXM45

Syntax

dspnodalcongcntr

Related Commands

None

Attributes

Log: nolog State: active, standby Privilege: ANYUSER

Example

Display the node-level congestion thresholds.

```
Geneva.7.PXM.a > dspnodalcongcntr
Parameter             Value         thresh1  thresh2
============================================================
setupRx                   0            90     100
statenqRx                 0           100     200
connpending               0           400     500
incompljournal            0             5

Parameter             Value          Mild    Medium    Severe
============================================================
vsiqdepth               0              5        10        20

Geneva.7.PXM.a >
```

dspnodalcongflags

Display Nodal Congestion Flags

The **dspnodalcongflags** command displays congestion flags maintained at the CCM at the node level.

Cards on Which This Command Runs

PXM45

Syntax

dspnodalcongflags

Related Commands

dspintfcongflags

Attributes

Log: nolog State: active, standby Privilege: ANYUSER

Example

Display the node-level congestion flags.

```
Geneva.7.PXM.a > dspnodalcongflags
Parameter                 Value
===================================
memflg                    FALSE
ipcflg                    FALSE
sarflg                    FALSE
remstatenqflg             FALSE
incompjournalingflg       FALSE
setupflg                  FALSE
connpendingflg            FALSE
connswitchoverflg         FALSE
conntrfparmflg            FALSE
conntrfabrflg             FALSE

Nodal Congestion Action Flags are

Parameter                 Value
===================================
dropsetupflg              FALSE
dropestabflg              FALSE
queuerel                  FALSE
markcallsforrelflag       FALSE
pacevsiresyncflg          FALSE
pacestatenqflg            FALSE
speedjournalflg           FALSE
pacepnniroutecalflg       FALSE
lowersetupthflg           FALSE

Geneva.7.PXM.a >
```

dspnodalcongth

Display Nodal Congestion Thresholds

Displays configuration of the nodal congestion thresholds.

Cards on Which This Command Runs

PXM45

Syntax

`dspnodalcongth`

Output Description

setuphi	The number of connection setup messages per second, above which the set-up messages have congested the node. Range: 1–105 calls per second Default: 100 calls per second
statenqlo	The number of status enquiry messages per second, below which the status enquiry congestion condition is dropped. Range: 1–500 calls per second Default: 100 calls per second
statenqhi	The number of connection setup messages per second, above which the status enquiries have congested the node. Range: 1 –500 calls per second Default: 200 calls per second
connpendlo	The aggregate number of connections in the establishment phase, below which establishment congestion is dropped. Range: 1–1000 connections Default: 400 connections
connpendhi	The aggregate number of connections in establishment phase above which the establishment congestion sets in Range: 1–1000 connections Default: 500 connections
incompjour	The number of incomplete journaling cycles that must be exceeded to increase the journaling speed. Range: 1–10 cycles Default: 5 cycles

vsiqmild	The VSI Q depth above which the VSI master is mildly congested. This threshold is represented as a percentage of the VSI master-slave communication window. This threshold applies to all the interfaces on the node. Range: 1–175 Default: 5
vsiqmedium	The VSI Q limit above which the VSI master is congested at a medium level. This threshold is represented as a percentage of VSI master-slave communication window size. This threshold applies to all the interfaces on the node. Range: 1–175 Default: 10
vsiqsevere	The VSI Q depth above which the VSI master is severely congested; This threshold is represented as a percentage of VSI master-slave communication window size.This threshold applies to all the interfaces on the node. Range: 1–175 Default: 20

Related Commands

cnfnodalcongth

Attributes

Log: nolog	State: active, standby	Privilege: SUPER_GP

Example

Display the current node-level congestion thresholds.

```
8850_SF.7.PXM.a > dspnodalcongth
Parameter          Value      Unit
=================================
setuphi(prov)       180       cps
setuphi(curr)       180       cps
statenqlo           100       cps
statenqhi           200       cps
connpendinglo       400       messages
connpendinghi       500       messages
incompjournalhi       5       cycles
vsiqdepthmild         5       multiplier
vsiqdepthmedium      10       multiplier
vsiqdepthsevere      20       multiplier

8850_SF.7.PXM.a >
```

dsppingatmaddr

Display Ping ATM Address

Display the default ping ATM address.

Cards on Which This Command Runs

PXM45

Syntax

dsppingatmaddr

Related Commands

dsppingatmaddr, **aesa_ping**

Attributes

Log: nolog	State: active	Privilege: ANYUSER

Example

Display the ATM ping address.

```
Geneva.7.PXM.a > dsppingatmaddr
47.0091.8100.0000.0030.ff0f.ef38.0030.ff0f.ef38.99
length: 160    type: unknown       proto: unknown
scope: 0       plan: unknown       redistribute: false


Geneva.7.PXM.a >
```

dsppncon

Display Connection—display information about an active SVC or SPVC.

The information that **dsppncon** displays corresponds to the objects described in the **portCallTable** and **portAbrCallTable** MIBs. This command can be issued at any node in the route. Refer to the example for the contents of the command output.

Cards on Which This Command Runs

PXM45

Syntax

dsppncon *<portid> <vpi> <vci>*

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
vpi	VPI for the call. Default: 0
vci	VCI for the call. If you do not specify a VCI, the connection is a virtual path connection. Default: 0

Related Commands

dsppncons

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display VPI/VCI 100 100 on port 1.5.

```
mpgses1.2.PXM.a > dsppncon 1.5 100 100

 Port :            1.5 VPI :    100  VCI :    100
 CallRef:      116  CallRefFlag:   0  CallLeafRef :        0
 Calling-address: 47.009181000000003071f80e4a.000000010500.00
 Calling-subaddress #1: N/A
 Calling-subaddress #2: N/A
 Called-address: 47.009181000000003071f80e49.000000010500.00
 Called-subaddress #1: N/A
 Called-subaddress #2: N/A
 OE Port :            1.7 OE VPI :      1  OE VCI :     39
 OE CallRef:       39  OE CallRefFlag:   0
 OAM-Type : Not an OAM Endpoint
 Connection-type : SPVC   Cast-type : point-to-point   Bearer-class :BCOBX
 Service-category :CBR   Call-clipping-susceptibility:no
 Tx conformance :CBR.1  Rx conformance :CBR.1
 Tx pcr :         50          Rx pcr :         50
 Tx scr :    N/A Rx scr :    N/A
 Tx mbs :    N/A Rx mbs :    N/A
 Tx cdvt : 250000
 Tx frame-discard-option :disable  Rx frame-discard-option :disable
 Max ctd :    N/A
 Max Tx cdv :    N/A    Max Rx cdv :    N/A

Max Tx clr :    N/A    Max Rx clr :    N/A
 NCCI value: 47 00 91 81 00 00 00 00 30 71 f8 0e 4a 00 30 71 f8 0e 4a 01 00 01
```

dsppncons

Display Port Connections

Displays a summary of SVCs and SPVCs on one port or all ports. The default is all connections on all ports because all parameters are optional. You can specify a starting VPI or VCI to begin a range of connections. In addition, you can display one of two *types*: point-to-point or point-to-multipoint.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppncons
[-port portid]
[-vpi starting-vpi]
[-vci starting-vci]
[-type {p2p | p2mp}]
```

Syntax Description

-port	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
-vpi	Starting VPI.
-vci	Starting VCI.
-type	Causes the system to display only point-to-point or point-to-multipoint. Without this option, the system displays both types. No default value exists if you include the keyword **type** on the command line, you must enter one of the possible entries. • “p2p” for point-to-point • “p2mp” for point-to-multipoint

Related Commands

dsppncon

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
Geneva.7.PXM.a > dsppncons

Port       VPI  VCI       CallRef X-Port       VPI  VCI   CallRef  Type OAM-Type
1:1.2:2     1   100       33       1:1.2:2      1   101      32    PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     1   101       32       1:1.2:2      1   100       33   PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     2   200       34       1:1.6:6      0    49   8388609  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.6:6     0    49  8388609       1:1.2:2      2   200        34  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00

Geneva.7.PXM.a >
```

dsppnconstats

Display Connection Statistics

Displays connection statistics for a port. Refer to the section, for a description of the items in the display.

Cards on Which This Command Runs

PXM45

Syntax

dsppnconstats <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

clrpnconstats

Attributes

Log: nolog	State: active	Privilege: ANYUSER

Examples

```
SanJose.7.PXM.a > dsppnconstats 4:1.1:11

 Call Statistics for 4:1.1:11
 Incoming Call Attempts:     209           Outgoing Call Attempts:       8

 Incoming Call Success:      6             Outgoing Call Success:        0

 Incoming Call Failures:     0             Outgoing Call Failures:       209

 Incoming Filtering Failures:0             Outgoing Filtering Failures : 0

 Incoming Routing Failures:  0             Outgoing Routing Failures :   209

 Incoming CAC Failures:      0             Outgoing CAC Failures :       0

 Incoming Timer Failures:    0             Outgoing Timer Failures :     0

 Incoming Crankback Failures:0             Outgoing Crankback Failures : 0

SanJose.7.PXM.a >
```

Output Description for dsppnconstats

Incoming Call Attempts	The number of incoming signaling messages—Setup and AddParty—received on this port for call establishment.
Incoming Call Success	The number of incoming signaling messages—Connect and AddPartAck—received on this port, which indicates successful call establishment.
Incoming Call Failures	The number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port.
Incoming Call Filtering Failures	The number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed the address filtering on this port.
Incoming Routing Failures	The number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there was no route to the destination.
Incoming CAC Failures	The number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there were not enough resources as requested in the traffic parameters of the call.
Incoming Timer Failures	The number of signaling timers that timed out on incoming point-to-point and point-to-multipoint SVC/SPVC calls received on this port.
Incoming Crankback Failures	The number of crankback IEs received on this port for incoming point-to-point and point-to-multipoint SVC/SPVC call attempts.
Outgoing Call Attempts	The number of outgoing signaling messages—Setup and AddParty—sent from this port for call establishment.
Outgoing Call Success	The number of outgoing signaling messages—Connect and AddPartAck—sent from this port, which indicates successful call establishment.
Outgoing Call Failures	The number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port.
Outgoing Call Filtering Failures	The number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed the address filtering on this port.
Outgoing Routing Failures	The number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because no route existed to the destination.
Outgoing CAC Failures	The number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because not enough resources existed to meet the request in the traffic parameters of the call.

Outgoing Timer Failures	The number of signaling timers that timed out on outgoing point-to-point SVC/SPVC calls sent from this port.
Outgoing Crankback Failures	The number of crankback information elements sent from this port for outgoing signaling release messages.

dsppnctlvc

Display Control VC Parameters

The command lets you display the bandwidth parameters for the control VCs on the port. These bandwidth parameters result from the use of the (optional) **cnfpnctlvc** command.

Note To see details about the VCs that support ILMI, use **dsppnilmi**.

Cards on Which This Command Runs

PXM45

Syntax

dsppnctlvc <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfpnctlvc

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the configuration of control VCs on port 10:1.1:1.

```
Geneva.7.PXM.a > dsppnctlvc 10:1.1:1

vc type = pnnircc
service category : sig                       PCR :             906
SCR :              453                       MBS :             171


vc type = sscop      Parameter = Provisioned
service category : sig                       PCR : Not Provisioned
SCR :  Not Provisioned                       MBS : Not Provisioned


vc type = sscop      Parameter = Operational
service category : cbr                       PCR :               0
Geneva.7.PXM.a >
```

dsppnport

Display PNNI Port—displays PNNI port information.

The **dsppnport** command shows dynamic, operational data rather than just the configuration data you would see by executing **dspport** on a service module.

The total number of connections that the **dsppnport** shows does not include control VCs. The types of control VCs are:

- Service-specific connection-oriented protocol (SSCOP)
- PNNI routing control channel (PNNI-RCC)
- Interim local management interface (ILMI) but only if ILMI is enabled

To see the connection counts that include control VCs, use the commands that operate on the CLI of the service modules (**dspln**, **dsppart** or **dsprscprtn**, **dspcd**, or **dspport**).

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnport <portid>
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

dspnports, **addpnport**, **delpnport**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

First determine if connections exist on port 3:1.1:1 by executing **dspcons**. One connection exists, and the display shows the view from each port (3:1.1:1 and 2:2.2:1). Display port 2:2.2:1 then port 3:1.1:1. Note the differences in the display when you specify the master-end port and the slave-end port. Also, note that the interface type is UNI 3.1.

The **dsppnport** display shows a combination of user-configured and dynamic details, as follows:

- Configuration details such as the type and version of the interface (UNI 3.1, for example), minimum and maximum VPIs for SPVCs, and minimum and maximum VPIs and VCIs for SVCs.
- A system-generated logical number that maps to the physical portID. The label these fields is "Logical ID" and "Port," respectively. The values in this example are 16979969 for logical ID and 3:1.1:1 for port. Some PNNI commands require you to provide the logical ID, and **dsppnport** is one command that can provide it.
- Dynamic information such as:
  - Status of the port
  - The number of point-to-point and point-to-multipoint connections
  - The numbers of configured and active of SPVCs and SVCs

```
8850_NY.7.PXM.a > dspcons

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
2:2.2:1            10 100    3:1.1:1            10 100     FAIL        MASTER
Local  Addr: 47.00918100000000036b5e30cd.000001021801.00
Remote Addr: 47.00918100000000036b5e30cd.000001031801.00
3:1.1:1            10 100    2:2.2:1            10 100     FAIL        SLAVE
Local  Addr: 47.00918100000000036b5e30cd.000001031801.00
Remote Addr: 47.00918100000000036b5e30cd.000001021801.00

8850_NY.7.PXM.a > dsppnport 2:2.2:1

Port:               2:2.2:1          Logical ID:       n/a
IF status:          provisioning     Admin Status:     up

8850_NY.7.PXM.a > dsppnport 3:1.1:1

Port:               3:1.1:1          Logical ID:       16979969
IF status:          up               Admin Status:     up
UCSM:               enable
Auto-config:        enable           Addrs-reg:        enable
IF-side:            network          IF-type:          uni
UniType:            private          Version:          uni3.1
PassAlongCapab:     n/a
Input filter:       0                Output filter:    0
minSvccVpi:         0                maxSvccVpi:       4095
minSvccVci:         35               maxSvccVci:       65535
minSvpcVpi:         1                maxSvpcVpi:       4095

       #SpvcCfg:  #SpvcActive:  #SpvpCfg:  #SpvpActive:
p2p :  1          0             0          0
p2mp:  0          0             0          0
       #Svcc:     #Svpc:        Total:
p2p :  0          0             0
p2mp:  0          0             0
                                Total:     0

8850_NY.7.PXM.a >
```

dsppnportcac

Display Port Call Admission Control

Displays CAC policy parameters for the port as configured by **cnfpnportcac**. For a list of the displayed items, see the Example section. For a description of these items, see the **cnfpnportcac** description.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportcac <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfpnportcac

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display the CAC parameters for port 3:2.2:4.

```
8850_NY.7.PXM.a > dsppnportcac 3:2.2:4

                  cbr:       rt-vbr:       nrt-vbr:          ubr:          abr:
        sig:
bookFactor:       100%          100%           100%          100%          100%
        100%
maxBw:       100.0000%     100.0000%      100.0000%     100.0000%     100.0000%
   100.0000%
minBw:         0.0000%       0.0000%        0.0000%       0.0000%       0.0000%
   100.0000%
maxVc:            100%          100%           100%          100%          100%
        100%
minVc:              0%            0%             0%            0%            0%
          1%
maxVcBw:           0             0              0             0             0
          0
```

dsppnportcc

Display Port Call Control

This command lets you display the call control parameters for a logical port. See **cnfpnportcc** for a description of applicable parameters.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportcc <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfpnportcc

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Examples

Display the call control configuration for port 3:1.1:1.

```
8850_NY.8.PXM.a > dsppnportcc 3:1.1:1
maxP2mpRoot:             1000
maxP2mpLeaf:             4095
svc blocking option:        no
spvc blocking option:       no

8850_NY.8.PXM.a >
```

dsppnportidmaps

Display Port ID Maps—display physical port ID to logical port ID mapping.

The **dsppnportidmaps** command lets you display the mapping of physical port identifiers (portIDs) to logical port identifiers. The purpose of having logical port IDs is that some command require the logical port ID. The displayed information consists of:

- Physical port ID in the format *slot*[:*subslot*].*port*[:*subport*]
- Logical port ID in decimal format
- Logical port ID in hexadecimal format
- Operational state of the port

Cards on Which This Command Runs

PXM45

Syntax

dsppnportidmaps

Syntax Description

This command takes no parameters.

Related Commands

none

Attributes

Log: nolog	State: active, standby	Privilege: SUPER_GP

Example

Display the mapping of physical port IDs to the logical IDs on the switch. Note that the switch has not generated a logical ID for port 2:2.2:1 because the port is still in the provisioning state.

```
8850_NY.7.PXM.a > dsppnportidmaps

Port Id         Logical ID (Dec)  Logical ID (Hex)  OperStatus

7.35            17251107          1073b23           up
7.36            17251108          1073b24           up
7.37            17251109          1073b25           up
7.38            17251110          1073b26           up
1:2.1:1         16848897          1011801           up
2:2.2:1         n/a               n/a               provisioning
3:1.1:1         16979969          1031801           up

8850_NY.7.PXM.a >
```

dsppnportloscallrel

Display PNNI Port Loss of Signal Call Release

This command displays the enable status of the LOS call release feature. See **cnfpnportloscallrel** for a description of this feature.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportloscallrel <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfpnportloscallrel

Attributes

Log: nolog	State: active, release	Privilege: ANYUSER

Example

First, confirm that LOS call release is disabled on port 3:1.1:1. Enable it, then confirm that it is enabled.

```
8850_NY.8.PXM.a > dsppnportloscallrel 3:1.1:1
Call release on Los :disabled
8850_NY.8.PXM.a > cnfpnportloscallrel 3:1.1:1 yes
8850_NY.8.PXM.a > dsppnportloscallrel 3:1.1:1
Call release on Los :enabled
```

dsppnportrange

Display Port Range

Displays ATM VPI/VCI range only for the port configured by **cnfpnportrange**. **dsppnport** displays the operational values.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportrange <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfpnportrange, dsppnport

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
Geneva.7.PXM.a > dsppnportrange
minSvccVpi: 0   maxSvccVpi:  4095
minSvccVci: 32  maxSvccVci: 65535
minSvpcVpi: 1   maxSvpcVpi:  4095

Geneva.7.PXM.a >
```

dsppnportrsrc

Display Port Resources

Displays available bandwidth and channels on the port for all service classes.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportrsrc <*portid*>

Syntax Description

portid	See PNNI Format, page 6-4. The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].

Related Commands

dsppnports

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display the available resources on port 11:1.1:11.

```
GGeneva.7.PXM.a > dsppnportrsrc 11:1.1:11
                                       cbr:     rt-vbr:     nrt-vbr:       ubr
:      abr:    sig:
Maximum Tx Cell Rate (cells/sec):      351707      351707       351707     35170
7    351707    353207
Maximum Rx Cell Rate (cells/sec):      351707      351707       351707     35170
7    351707    353207
Min Guarant Tx Cell Rate (cells/sec):       0           0            0
0         0      1500
Min Guarant Rx Cell Rate (cells/sec):       0           0            0
0         0      1500
Minimum Cell Loss Ratio Tx :               10           8            6
6         6         8
Minimum Cell Loss Ratio Rx :               10           8            6
6         6         8
Available Tx Cell Rate (cells/sec):    344035      344035       344035     34403
5    344035    344035
Available Rx Cell Rate (cells/sec):    344035      344035       344035     34403
5    344035    344035
# of Available Tx Channels:             28903       28903        28903      2890
3     28903     28903
# of Available Rx Channels:             28903       28903        28903      2890
3     28903     28903
```

dsppnports

Display Ports

The **dsppnports** command displays status for all logical ports. If you do not identify a particular type of interface, the display shows all interface types.

In brief, the display consists of:

- A summary of connections
- The logical number associated with the PNNI port number (a format that some commands require)
- A status summary for each port, including the number of connections on each port (excluding control VCs)

For details on the connection summaries, see the section, "Description of the dsppnports Connection Summaries."

Note The total number of connections that the display shows does not include control VCs. The types of control VCs are SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). To see the connection counts that include control VCs, use the commands that operate on the CLI of the service modules (**dspln**, **dsppart** or **dsprscprtn**, **dspcd**, and **dspport**).

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnports [uni | nni | enni]
```

Syntax Description

uni	Show status for only UNI ports.
nni	Show status for only NNI ports.
enni	Show status for only ENNI ports.

ILMI States

The display includes ILMI status, as follows:

Disable	Protocol is not enabled on this port.
NotApplicable	This port is not accessible due to hardware-related conditions.
LostConnectivity	Protocol on listening port is not enabled.

EnableNotUp	This port is not accessible due to hardware.
UpAndNormal	This port is physically up, and the protocol is enabled.

Related Commands

dsppnportrsrc, **dsppnport**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Description of the dsppnports Connection Summaries

This section contains the following:

1. A list identifies all fields in the connection summary part of the display.
2. A simple network description illustrates how an SPVC in a three-node network would appear in the summary part of the display.
3. A description of the summary for a DAX connection follows the SPVC explanation.

The list of summary fields follows:

- *Total* point-to-point or point-to-multipoint connections
  - SVCC, switched virtual channel connections
  - SVPC, switched virtual path connections
  - SPVCD, semi-permanent virtual channel DAX connections
  - SPVPD, semi-permanent virtual path DAX connections
  - SPVCR, active (routed) semi-permanent virtual circuits
  - SPVPR, active (routed) semi-permanent virtual paths
  - Total of all the preceding types
- *Configured SPVC endpoints* for either point-to-point and point-to-multipoint connections
  - SPVCD, semi-permanent virtual channel DAX connections
  - SPVPD, semi-permanent virtual path DAX connections
  - SPVCR, active (routed) semi-permanent virtual circuits
  - SPVPR, active (routed) semi-permanent virtual paths
  - Totals for the preceding types

For an example SPVC, refer to Figure 6-1. An SPVC's master endpoint is on a UNI on Node 1. The slave endpoint is on a UNI on Node 3. The SPVC traverses a via node, Node 2. If you run the **dsppnports** command on Node 1, the display gives the following information in the four parts of the summary:

- Number of connections: SpvcR = 1
- Number of configured endpoints: SpvcR = 1
- Total of 1 connection, 2 endpoints (1 configured, 1 active intermediate on the via node).

If you run the **dsppnports** command on via Node 2, the display gives the following information in the four parts of the summary:

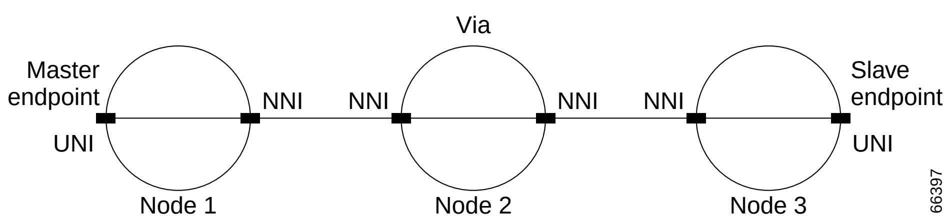
- Number of connections: Svcc = 1
- Number of configured endpoints: 0
- Total of 1 connection, 2 endpoints (2 active intermediate)

Next, a DAX connection has both endpoints on the same switch (the DAX does not appear in Figure 6-1). The summary information for a DAX connection would appear as follows:

- Number of connections: SpvcD = 1
- Number of configured endpoints: SpvcD = 2
- Total of 1 connections, 2 endpoints (2 configured)

In summary, each active connection has two endpoints. Configured endpoints translate to the endpoints added through the addcon command. (shown as spvc on AXSM). Active, intermediate endpoints translate to the NNI sides for and SPVC or both sides for a pure SVC (shown as an SVC on an AXSM).

Figure 6-1 An SPVC With Endpoints and a Via Node



Example

Display all PNNI logical ports on the switch. UNI ports 7.35, 7.36, 7.37, and 7.38 are reserved for BITS clock sources on the PXM45 UI card. In reality, only 7.35 and 7.36 are meaningful.

```
M8850_NY.7.PXM.a > dsppnports
Summary of total connections
(p2p=point to point,p2mp=point to multipoint,SpvcD=DAX spvc,SpvcR=Routed spvc)
Type   #Svcc:   #Svpc:   #SpvcD:  #SpvpD:  #SpvcR:  #SpvpR:  #Total:
p2p:   0        0        0        0        0        0        0
p2mp:  0        0        0        0        0        0        0

                         Total=      0/50000

Summary of total configured SPVC endpoints
Type   #SpvcR   #SpvpR   #SpvcD   #SpvpD   Total
p2p:   0         0         2         0         2
p2mp:  0         0         0         0         0
                         Total=2

Per-port status summary

PortId      LogicalId    IF status      Admin status  ILMI state        #Conns

7.35         17251107    up             up            NotApplicable     0

7.36         17251108    up             up            NotApplicable     0

7.37         17251109    up             up            NotApplicable     0

7.38         17251110    up             up            NotApplicable     0

1:2.1:1      16848897    up             up            UpAndNormal       0

2:2.2:1      16914433    provisioning   up            NotApplicable     0

3:1.1:1      16979969    down           up            Disable           0

3:1.2:2      0           provisioning   down          NotApplicable     0

6:1.1:1      17176577    up             up            Disable           0

M8850_NY.7.PXM.a >
```

dsppnportsig

Display Port Signaling—displays signaling parameters on a port.

The **dsppnportsig** command displays the ATM signaling parameters as configured by **cnfpnportsig**.

Cards on Which This Command Runs

PXM45

Syntax

dsppnportsig <*portid*>

Syntax Description

portid	See PNNI Format, page 6-4. The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].

Related Commands

cnfpnportsig

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Examples

Display signaling for port 3:1.1:1 then port 2:2.2:1. Note the interface type for 3:1.1:1 is UNI, and the UNI version is 3.1. On port 2:2.2:1, the interface type is NNI, and the version is IISP 3.1.

```
8850_NY.8.PXM.a > dsppnportsig 3:1.1:1

provisioned IF-type: uni   version:     uni3.1
sigType:  private          side:        network
addrPlan:  aesa
VpiVciAllocator:  n/a      HopCounterGen:  n/a
PassAlongCapab:  n/a
sigVpi:          0         sigVci:             5
rccVpi:        n/a         rccVci:           n/a

8850_NY.8.PXM.a > dsppnportsig 2:2.2:1

provisioned IF-type: nni   version:     iisp31
sigType:  private          side:        network
addrPlan:  aesa
VpiVciAllocator:  n/a      HopCounterGen:  n/a
PassAlongCapab:  enable
sigVpi:          0         sigVci:             5
rccVpi:        n/a         rccVci:           n/a

8850_NY.8.PXM.a >
```

dsppnsysaddr

Display PNNI Port System Addresses—displays the PNNI system address table.

The **dsppnsysaddr** command displays addresses in the system address table. The system address table only contains static addresses.

Note This command does not belong to the RA module.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppnsysaddr
[ ilmi | uni | static | host | all ]
```

Syntax Description

ilmi	Display all of the ilmi addresses in the peer group.
uni	Display all of the uni addresses in the peer group.
static	Display all of the static addresses in the peer group.
host	Display all of the host addresses in the peer group.
all	Display all of the addresses. This is the default.

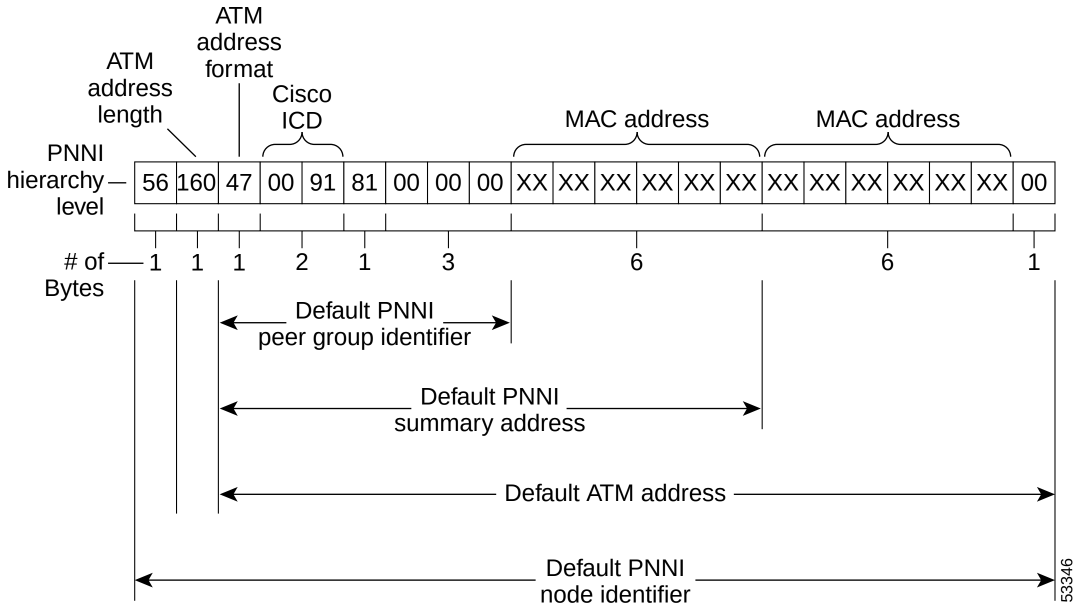
Display Contents

The ATM address, the address prefixes, and the peer group identifier share some default field values, as shown in Figure 6-2.

The following parameters are displayed for each node.

ATM Address (displayed but not labeled)	Display the PNNI node ATM address. This is a 20-byte, formatted hexadecimal string. Like all PNNI addresses, identifiers, and prefixes, this value is portrayed as a string of hexadecimal “nibbles.” One or several pairs of nibbles entail each parameter field. Default: Figure 6-2 shows the factory-set default.
Type	Display the type of address that you specified in the command line. Range: ilmi, uni, static, host
Port id	The PNNI logical port identifier. Range: 1–2147483648

Figure 6-2 Cisco Factory-shipped Defaults for PNNI Peer Group Identifier, PNNI Summary Address, ATM Address, and PNNI Node Identifier



Related Commands

None

Attributes

Log: log State: active, standby Privilege: ANYUSER

Examples

Display addresses in the System Address Table. The first command entry includes the option **all**, so **dsppnsysaddr** displays all addresses in the peer group.

```
Geneva.7.PXM.a > dsppnsysaddr all


47.0091.8100.0000.0030.9409.f1f1.0030.9409.f1f1.01/160
Type:     host     Port id:   17251106


47.0091.8100.0000.0030.9409.f1f1.0030.9409.f1f1.99/160
Type:     host     Port id:   17251106


47.0091.8100.0000.0030.9409.f1f1.0011.1010.0000.01/160
Type:     host     Port id:   17251106


47.0091.8100.0000.0030.9409.f1f2.0000.0104.180b.00/160
```

```
Type:     host     Port id:   17251106


47.0091.8100.0000.0030.9409.f1f2.0000.0104.180c.00/160
Type:     host     Port id:   17251106


47.1111.1111.1111.1111.1111.1111.0000.0000.0000/152


Type:      uni     Port id:   17045515


47.1111.1111.1111.1111.1111.1111.0000.0000.0000/152
Type:      uni     Port id:   17045516

Geneva.7.PXM.a >
```

The following example includes the option “host,” so **dsppnsysaddr** displays only the host addresses in the peer group.

```
Geneva.7.PXM.a > dsppnsysaddr host

47.0091.8100.0000.0030.ff0f.ef38.0000.010b.180b.00/160
Type:     host     Port id:   17251106

47.0091.8100.0000.0030.ff0f.ef38.0000.010d.180b.00/160
Type:     host     Port id:   17251106

47.0091.8100.0000.0030.ff0f.ef38.1111.1101.0001.01/160
Type:     host     Port id:   17251106


Geneva.7.PXM.a >
```

dspsig

Display Signaling

Displays the configuration of the signaling timers for the port.

Cards on Which This Command Runs

PXM45

Syntax

`dspsig` `<portid>`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfsig

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display the signaling timers and crankback maximum for port 4:1.1:11.

```
SanJose.7.PXM.a > dspsig 4:1.1:11

Signaling Timers for port : 4:1.1:11

Timer           Value(secs)
-----           -----------
t301            180
t303            4
t308            30
t310            10
t316            90
t317            60
t322            4
t397            180
t398            4
t399            14

Max Crankback:  3

SanJose.7.PXM.a >
```

dspsigdiag

Display Signaling Diagnostic

Displays the configured filter entries and the collection call records for the ATM signaling diagnostics.

Cards on Which This Command Runs

PXM45

Syntax

dspsigdiag {**filter | rec** | **status**}
[*index*]

Syntax Description

filter	Display the information in the filter table.
rec	Display the call failure records.
status	Display global diagnostics status.
index	This field can be used with the option **filter** or **rec**. If used with **filter**, the configuration of the specified indexed filter entry display. If you don't specify an index, configuration of all filter entries display. If this field is used with **rec**, then all of the records filtered for the specified indexed filter entry display.

Related Commands

cnfsigdiag, delsigdiag, dspsigstats, clrsigstats

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

Display the filter sets.

```
SanJose.7.PXM.a > dspsigdiag filter

F I L T E R   I N D E X    1
--------------------------------
Scope: internal, Cast Type: p2mp
Connection Kind: soft-vc
Service Category: CBR (Constant Bit Rate) UBR (Unspecified Bit Rate)
Clear Cause: 0, Initial TimerValue: 600
Max Records: 20,   NumMatches: 0,   Timer expiry: 600
Incoming Port: ATM0/0/1, Outgoing Port: ATM0/1/1
Calling Nsap Address:47.1111222233334444555566666.777788889999.00
Calling Address Mask:FF.FFFFFF000000000000000000.000000000000.00
Called Nsap Address:47.1111222233334444555566666.777788889999.01
Called Address Mask:FF.FFFFFF000000000000000000.000000000000.00
Status: active

F I L T E R   I N D E X    2
--------------------------------
```

Display the records associated with index 1.

```
SanJose.7.PXM.a > dspsigdiag rec 1

D I S P L A Y I N D E X    1
--------------------------------
Scope: internal, Cast Type: p2p, Conn Indicator: Setup Failure
Connection Kind:   switched-vc
Service Category: UBR (Unspecified Bit Rate)
Clear Cause: 0x29, Diagnostics: NULL
Incoming Port: ATM1/0/3, Outgoing Port:ATM0/1/3
Calling-Address: 47.00918100000000601100000.470803040506.00
Calling-SubAddr: NULL
Called-Address : 47.009181000000006083C42C01.750203040506.00
Called-SubAddr : NULL
Crankback Type : No Crankback
DTL's:
NodeId:56:160:47.00918100000000601100000.006083AB9001.00 Port: 0/1/3:2
NodeId:56:160:47.00918100000000603E7B4101.00603E7B4101.00 Port: 0/0/0:2
NodeId:56:160:47.009181000000006083C42C01.006083C42C01.00 Port: 0

D I S P L A Y I N D E X    2
--------------------------------

SanJose.7.PXM.a >
```

dspsigparm

Display Signaling Parameter

The **dspsigparm** command shows whether the switch is enabled to perform frame discard on AAL5 cells. The **cnffdonaal5** command lets you enable or disable frame discard for AAL5 cells. The default is enabled.

Cards on Which This Command Runs

PXM45

Syntax

dspsigparm

Syntax Description

This command takes no parameters.

Related Commands

cnffdonaal5

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Determine whether frame discard for AAL5 cells is enabled on this switch.

```
SanJose.7.PXM.a > dspsigparm
Global Signaling Parameters
============================
Frame Discard on AAL5 IE: yes

SanJose.7.PXM.a >
```

dspsigstats

Display Signaling Statistics

Displays signaling statistics for a port. See the Example section for a list of the types of statistics.

Cards on Which This Command Runs

PXM45

Syntax

`dspsigstats <portid>`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfsigdiag, delsigdiag, dspsigdiag, clrsigstats

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

Display signaling statistics for port 1:2.1:1.

```
8850_NY.8.PXM.a > dspsigstats 1:2.1:1

 Signaling Statistics for 1:2.1:1
 Message                          Rcv           Xmt
 -------                          ---           ---
 Call Proceeding                    0             0
 Connect                            0             0
 Connect Ack                        0             0
 Setup                              0             0
 Release                            0             0
 Release Complete                   0             0
 Add Party                          0             0
 Add Party Ack                      0             0
 Add Party Rej                      0             0
 Drop Party                         0             0
 Restart                            0             0
 Restart Ack                        0             0
 Status                             0             0
 Status Enquiry                     0             0
 Alerting                           0             0
 Notify                             0             0
 Progress                           0             0

Last Cause/Diag/Crankback
 --------------------------
 Cause                              0
 Diagnostic                         0       0       0       0
 Src Crankback port count           0
```

Display the SPVC address first on port 1:1.1:1 then on port 1:1.4:4.

```
p2spvc5.7.PXM.a > dspspvcaddr 1:1.1:1

Interface Id      Soft VC Address(es)
------------      -------------------
     1:1.1:1     47.0091.8100.0000.0010.7be9.2f6d.0000.0101.1801.00

p2spvc5.7.PXM.a > dspspvcaddr 1:1.4:4

Interface Id      Soft VC Address(es)
------------      -------------------
     1:1.4:4     47.0091.8100.0000.0010.7be9.2f6d.0000.0101.1804.00
```

dspspvcprfx

Display SPVC Prefix

The **dspspvcprfx** command displays the prefix for the switch-level SPVC address. The switch comes with a default SPVC prefix, and you can modify the prefix by executing **cnfspvcprfx**. See description of **cnfspvcprfx** for more details.

Cards on Which This Command Runs

PXM45

Syntax

```
dspspvcprfx
```

Syntax Description

This command takes no parameters.

Related Commands

cnfspvcprfx, **dspspvcaddr**

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
MGX8850.7.PXM.a > dspspvcprfx
SPVC Node Prefix: 47.00918100000100001a531c2a
```

dspsscop

Display SSCOP—display SSCOP details for a PNNI port.

The **dspsscop** command displays information about the state of the link on the port, status inquiry and response timers, and statistics.

Cards on Which This Command Runs

PXM45

Syntax

`dspsscop <portid>`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfsscop, disablesscop, dspsscopstats

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
SanJose.7.PXM.a > dspsscop 4:1.1:11

SSCOP details for interface: 4:1.1:11

   Current State = enabled, Current Link State = Established State,
SSCOP version = 3.1
   Send Sequence Number: Current = 153,  Maximum = 183
   Send Sequence Number Acked = 153
   Rcv Sequence Number: Lower Edge = 155, Upper Edge = 155, Max = 185
   Poll Sequence Number = 427, Poll Ack Sequence Number = 427
   Vt(Pd) = 0   Vt(Sq) = 1
   Timer_IDLE = 10 - Active
   Timer_CC = 1 - Inactive
   Timer_POLL = 1 - Inactive
   Timer_KEEPALIVE = 5 - Inactive
   Timer_NO-RESPONSE = 30 - Inactive
   Timer_T309 = 10 - Inactive
   Max CC = 10
   Send Window = 30
   Recv Window = 30
   Current Retry Count = 33932, Maximum Retry Count = 10
   AckQ count = 0, RcvQ count = 0, TxQ count = 0
   AckQ HWM = 1,  RcvQ HWM = 0, TxQ HWM = 1

      Statistics -
      Pdu's Sent = 1011, Pdu's Received = 1004, Pdu's Ignored = 0
      Begin = 1/4, Begin Ack = 0/1, Begin Reject = 0/0
      End = 0/0, End Ack = 0/0
      Resync = 0/0, Resync Ack = 0/0
      Sequenced Data = 155/153, Sequenced Poll Data = 0/0
      Poll = 426/427, Stat = 422/426, Unsolicited Stat = 0/0
      Unassured Data = 0/0, Mgmt Data = 0/0, Unknown Pdu's = 0
      Lack of credit = 0

SanJose.7.PXM.a >
```

dspsscopstats

Display SSCOP Statistics

Displays SSCOP statistics for the port.

Cards on Which This Command Runs

PXM45

Syntax

`dspsscopstats` `<portid>`

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

cnfsscop, disablesscop, dspsscop

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Examples

```
SanJose.7.PXM.a > dspsscopstats 4:1.1:11

SSCOP Statistics for interface: 4:1.1:11

      Pdu's Sent = 1045, Pdu's Received = 1037, Pdu's Ignored = 0
      Begin = 1/4, Begin Ack = 0/1, Begin Reject = 0/0
      End = 0/0, End Ack = 0/0
      Resync = 0/0, Resync Ack = 0/0
      Sequenced Data = 160/158, Sequenced Poll Data = 0/0
      Poll = 440/442, Stat = 436/440, Unsolicited Stat = 0/0
      Unassured Data = 0/0, Mgmt Data = 0/0, Unknown Pdu's =
 0
      Lack of credit = 0

SanJose.7.PXM.a >
```

dspsvcparm

Display SVC Parameters

Displays global SVC parameters for the node.

Cards on Which This Command Runs

PXM45

Syntax

dspsvcparm

Related Commands

cnfe164justify

Attributes

Log: nolog	State: active, standby	Privilege: ANYUSER

Example

```
SanJose.7.PXM.a > dspsvcparm
Global SVC parameters
============================
E164 Address Conversion Justification :left


SanJose.7.PXM.a >
```

dsptrapip

Display Trap IP

The **dsptrapip** command displays the switch trap IP address. The switch must have a trap IP assigned by the **cnftrapip** command.

Cards on Which This Command Runs

PXM45

Syntax

`dsptrapip`

Syntax Description

This command takes no parameters.

Related Commands

cnftrapip, **dsptrapmgr**, **addtrapmgr**, **deltrapmgr**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Assign and confirm a trap IP address.

```
SanJose.7.PXM > cnftrapip 172.27.27.184

SanJose.7.PXM > dsptrapip
Trap IP Address :172.27.27.184

SanJose.7.PXM >
```

tstpndelay

Test Port Delay

Initiates loopback test for a connection. If you provide both a VPI and VCI, the segment endpoint is an F5 flow endpoint (for VCCs). If you provide only a VPI, the segment endpoint is a F4 flow endpoint (for VPCs). This command displays the round trip delay in microseconds.

Use **cnfconsegep** to specify the segment endpoint if one does not already exist.

Cards on Which This Command Runs

PXM45

Syntax

```
tstpndelay <portid> <vpi> [vci]
[-direction {inbound | outbound}]
```

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.
vpi	VPI of the connection.
vci	VCI of the connection. Default: 0
-direction	Specifies the direction of loopback. **inbound:** endpoint towards the backplane **outbound:** endpoint departing the port Default: inbound

Related Commands

cnfconsegep

Attributes

Log: log	State: active	Privilege: SUPER_GP

uppnport

Up PNNI Port:

The **uppnport** command lets you put a UNI or NNI port into service (administratively “up” the port). The circumstances in which you would use the **uppnport** command are as follows:

- After you have downed the port for any reason by using the **dnpnport** command
- After you have pre-configured a port by using the **addpnport** command (see t**addpnport** description)

Cards on Which This Command Runs

PXM45

Syntax

uppnport <*portid*>

Syntax Description

portid	The *portid* is the PNNI physical port. The format is [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 6-4.

Related Commands

addpnport, **delpnport**, **dnpnport**, **dsppnport**, **dsppnports**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Check for any downed ports by executing **dsppnports**. Use **uppnport** to up any ports that are down. The output of **dsppnports** shows that only one user port exists (not the 7.3x ports, the first two of which are the BITS clock ports).

```
8850_NY.8.PXM.a > dsppnports
Summary of total connections
(p2p=point to point,p2mp=point to multipoint,SpvcD=DAX spvc,SpvcR=Routed spvc)
Type   #Svcc:   #Svpc:   #SpvcD:  #SpvpD:  #SpvcR:  #SpvpR:  #Total:
p2p:   0        0        0        0        0        0        0
p2mp:  0        0        0        0        0        0        0

                         Total=      0/50000

Summary of total configured SPVC endpoints
Type   #SpvcR   #SpvpR   #SpvcD   #SpvpD   Total
p2p:   0         0         0         0         0
p2mp:  0         0         0         0         0
                         Total=0

Per-port status summary

PortId      LogicalId    IF status      Admin status  ILMI state        #Conns

7.35         17251107    up             up            Undefined         0

7.36         17251108    up             up            Undefined         0

7.37         17251109    up             up            Undefined         0

7.38         17251110    up             up            Undefined         0

3:1.1:1      16979969    down           down          Disable           0

8850_NY.8.PXM.a >

SanJose.7.PXM.a > uppnport 3:1.1:1
```

Check the administrative status of 3:1.1:1.

```
8850_NY.8.PXM.a > dsppnport 3:1.1:1

Port:               3:1.1:1           Logical ID:       16979969
IF status:          up                Admin Status:     up
UCSM:               enable
Auto-config:        enable            Addrs-reg:        enable
IF-side:            network           IF-type:          uni
UniType:            private           Version:          uni3.1
PassAlongCapab:     n/a
Input filter:       0                 Output filter:    0
minSvccVpi:         0                 maxSvccVpi:       4095
minSvccVci:         35                maxSvccVci:       65535
minSvpcVpi:         1                 maxSvpcVpi:       4095
```

CHAPTER 7

SPVC and SVC Commands

This chapter describes the commands that apply to semi-permanent virtual circuits (SPVCs) and switched virtual circuits (SVCs). These commands allow you to add, delete, configure, display status, and specify statistics for these connections. The commands include the following areas:

- Adding a connection
- Modifying an existing connection
- Deleting a connection
- Display one connection or all connections
- Downing and upping a connection in the course of maintenance or troubleshooting

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num. conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 7-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node uses the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation*, Release 2.

Table 7-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–2	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 7-2 and Table 7-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 7-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 7-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 7-4.

Table 7-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 7-4 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1

- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

addcon

Add Connection

Adds a logical connection as an SPVC on a service module. The switch assigns a 20 octet NSAP address to the slave endpoint, which is sent back to the master and uniquely identifies the endpoint on the network. An AXSM front card can support a maximum of 64K SPVCs. This command does not apply to SVCs.

Before Adding a Connection

Before you can add an SPVC, the following tasks must have been completed:

1. The switch must have a network controller (see **addcontroller** description).
2. A physical line must be active. Use the **upln** command or the CiscoView application.
3. At least one logical port must exist on the active physical line. Use the **addport** command or the CiscoView application to create the port. If necessary, modify the port through **cnfport**.)
4. At least one resource partition must exist on the logical port. Use the **addrscprtn** (or **addpart**,) command or the CiscoView application. The resource partition should be associated with the controller you add in step 1.

Adding a Connection

Adding a connection requires you first to provision a slave endpoint. Subsequently, you again execute **addcon** to provision a master endpoint. The master endpoint of the connection initiates the routing of the call and can be viewed as the “calling” party. The slave endpoint is the called endpoint. The following are characteristics of this master-slave arrangement:

- When you add a slave endpoint, the system returns a *slave endpoint identifier*. You subsequently need to provide this slave endpoint identifier when specifying the master endpoint.
- When you add the master endpoint, you must provide the slave endpoint identifier. (A copy and paste operation is probably more convenient than writing down the slave endpoint identifier.) After you finish adding the master endpoint, the switch starts routing the connection.

To modify the bandwidth parameters or configure usage parameter control (UPC), use **cnfcon** for all service types. In addition, ABR connections require more configurable parameters for implementing closed loop control. Use **cnfabr** to configure the ABR parameters. (Note, however, that ABR VSVD is not supported in the current release.)

Usage Guidelines

The following sections detail the application of certain **addcon** parameters.

Traffic Parameters

Traffic parameters such as PCR, SCR, MBS are entered at both the master and slave endpoints for both the forward and reverse directions. Be sure that the value entered as “local” on one end is equal to the value entered as “remote” on the other end. For example, the *lpcr* on the slave endpoint should be same as the *rpcr* on the master endpoint and vice versa when you provision the connection at the other end. If you modify traffic parameters after creating an SPVC, you just modify them at either the master endpoint or the slave endpoint.

Traffic parameters such as CDV, CTD are entered at both the master and slave endpoints for both the forward and reverse directions. However, the values of these parameters entered at the slave end are ignored during call setup. Therefore, you can specify the *lcdv*, *rcdv*, *lctd* and *rctd* options at the master end only.

SCT Default Traffic Parameters

The Service Class Templates (SCTs) provide the default traffic parameters for the logical ports. The default traffic parameters are set to a fraction of the bandwidth available on the logical port. The SCT ID (*sctID*) and interface type (*ifType*) parameters that are specified using the **addport** command determine which default traffic parameters are used.

Note The default values of the SCTs cannot be changed using the Cisco WAN Manager (CWM).

Table 7-5 Default Traffic Parameters for AXSM

	PCR	SCR	MCR	ICR	MBS	MFS	CDVT
VSI-SIG	N/P	N/P	N/P	N/P	N/P	N/U	N/P
CBR.1	50	N/A	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-RT.1	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-RT.2	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-RT.3	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-nRT.1	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-nRT.2	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
VBR-nRT.3	50	50	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
UBR.1	50	N/A	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
UBR.2	50	N/A	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
ABR	50	N/A	50	50	dspmbsdft	N/U	dspcdvtdft
CBR.2	50	N/A	N/A	N/A	dspmbsdft	N/U	dspcdvtdft
CBR.3	50	N/A	N/A	N/A	dspmbsdft	N/U	dspcdvtdft

Table 7-6 Ranges for PCR, SCR, and MCR for Each Line Type

Parameter	Range
PCR	Minimum value of 7 cell/sec. Maximum depends on the physical line on which the interface is configured: for OC12:between 7 and 1412832 for OC3:between 7 and 353208 for T3:between 7 and 96000(PLCP), 104268(ADM) for E3:between 7 and 80000 for T1:between 7 and 3622 for E1:between 7 and 4528 Default : Taken from the SCT which was chosen for the virtual interface. The service type is used as an index in choosing a value of PCR. The default value of PCR in the SCT is defined as a percentage of the interface bandwidth.
SCR	Minimum value of 7 cells/sec. Maximum is limited to the PCR. Default: Taken from SCT as a percentage of PCR.
MCR	Same as SCR

Routing Parameters

The routing parameter *maxcost* (specified using -**mc** option) need to be entered at the master endpoint only. The values of the parameters entered at the slave end are ignored and during call set-up.

Frame Discard

For the parameter *frame discard* (specified using -**frame** option), you need to enter it at only the master endpoint. This parameters has no significance at the slave end.

For the MGX 8850 2.0 release, if you try to enable frame discard at the slave end point you will not get an error message. Nothing will happen, and frame discard will not take effect. In future releases, an error message will be displayed if you try to enable frame discard at the slave end point.

Local-Only Parameters

The parameters CDVT, stats enable, cc enable (specified using -**cdvt**, -**stat**, -**cc**) are significant only at the endpoint where you enter them. Therefore, they can be different at each end of the connection.

Cards on Which This Command Runs

AXSM

Syntax

```
addcon <ifNum> <vpi> <vci> <service type> <mastership>
[-slave <NSAP.vpi.vci>]
[-lpcr <local PCR>]
[-rpcr <remote PCR>]
[-lscr <local SCR>]
[-rscr <remote SCR>]
[-lmbs <local MBS>]
[-rmbs <remote MBS>]
[-cdvt <local CDVT>]
[-lcdv <local maxCDV>]
[-rcdv <remote maxCDV>]
[-lctd <local maxCTD>]
[-rctd <remote maxCTD>]
[-cc <OAM CC Cnfg>]
[-stat <Stats Cnfg>]
[-frame <frame discard>]
[-mc <maximum cost>]
```

Syntax Description

For the applicable parameters, the “local” end is the point at which you are provisioning the connection.

ifNum	The logical port number. The range for AXSM is 1–60. This *ifNum* corresponds to the *ifNum* added through the **addport** command. When adding an endpoint on a NNI port, make sure that PNNI signaling is disabled on the PXM45 (**cnfpnportsig** <portid> **-nniver none**).
vpi	Virtual path identifier value in the range 0–255 (UNI) or 0–4095 (NNI or VNNI). For VNNI, specify one VPI per port.
vci	Virtual connection identifier (VCI): • For a VCC on a UNI, the range is 1–4095. On an NNI or VNNI, the VCI range is 32–65535. For MPLS, the recommended minimum VCI is 35. • For a VPC, the *vci* is 0.

service type	Value in the range 1–12 to specify the service type: • 1=CBR1 (Constant Bit Rate 1) • 2=VBR1RT (Variable Bit Rate 1, Real Time) • 3=VBR2RT (Variable Bit Rate 2, Real Time) • 4=VBR3RT (Variable Bit Rate 3, Real Time) • 5=VBR1NRT (Variable Bit Rate 1, Non-Real Time) • 6=VBR2NRT (Variable Bit Rate 2, Non-Real Time) • 7=VBR3NRT (Variable Bit Rate 3, Non-Real Time) • 8=UBR1 (Unspecified Bit Rate 1) • 9=UBR2 (Unspecified Bit Rate 2) • 10=ABRSTD (Standard ABR—see **cnfabr** for ABR-specific parameters) • 11=CBR2 (Constant Bit Rate 2) • 12=CBR3 (Constant Bit Rate 3)
mastership	Value to specify the endpoint as master or slave: • 1 or ‘m’ specifies the master end. • 2 or ‘s’ specifies the slave end.
-slave	Keyword that precedes the slave-end identifier, an item you *enter* at the master end. This keyword is mandatory when you are adding a *master* endpoint (*mastership*=m or 1)**.**
-lpcr	Local Peak Cell Rate (PCR). Specifies the PCR from a local endpoint to a remote endpoint (3–5651328 cells per second). Peak Cell Rate is the maximum cell rate for the connection at any time.
-rpcr	Remote Peak Cell Rate (PCR). Specifies the PCR from a remote endpoint to a local endpoint (3–5651328 cells per second). Peak Cell Rate is the maximum cell rate for the connection at any time.
-lscr	Local Sustained Cell Rate (SCR). Specifies the SCR from a local endpoint to a remote endpoint (3–5651328 cells per second). Sustained Cell Rate is the maximum cell rate that a connection can sustain for long periods.
-rscr	Remote Sustained Cell Rate (SCR). Specifies the SCR from a remote endpoint to a local endpoint (3–5651328 cells per second). Sustained Cell Rate is the maximum cell rate that a connection can sustain for long periods.
-lmbs	Local Maximum Burst Size (MBS). Specifies the MBS from a local endpoint to a remote endpoint (1–5000000 cells). Maximum Burst Size is the maximum number of cells that can burst at the PCR and still be compliant.
-rmbs	Remote Maximum Burst Size (MBS). Specifies the MBS from a remote endpoint to a local endpoint (1–5000000 cells). Maximum Burst Size is the maximum number of cells that can burst at the PCR and still be compliant.

-cdvt	Local Cell Delay Variation Tolerance (CDVT). Specifies the CDVT from a local endpoint to a remote endpoint (1–5000000 microseconds). Cell Delay Variation Tolerance controls the time scale over which the PCR is policed. Note that no *remote* CDVT is necessary.
-lcdv	Local Cell Delay Variation (CDV). Specifies the CDV from a local endpoint to a remote endpoint (1–16777215 microseconds). Cell Delay Variation is the peak to peak cell delay variation expressed in microseconds.
-rcdv	Remote Cell Delay Variation (CDV). Specifies the CDV from a remote endpoint to a local endpoint (1–16777215 microseconds). Cell Delay Variation is the peak to peak cell delay variation expressed in microseconds.
-cc	Operations, administration, and maintenance continuity check (OAM CC): enter 1 to enable or 0 to disable. The default is 0. To provision continuity checking, you must enable this function at both ends of the connection, otherwise a connection alarm results. Therefore, even as you create the connection with this parameter, the connection goes into alarm until both ends of the connection are up.
-stat	Statistics collection: enter 1 to enable or 0 to disable. The default is 0. The Cisco WAN Manager tool collects statistics for a connection if you enable it here. Statistics collection is disabled for all connections by default. Statistics collection has an impact (which may not be significant) on the real-time response, especially for SVCs (which can be affected even though you do not add SVCs). Therefore, you should enable statistics collection for only the subset of connections that really warrants such a feature.

-frame Frame discard: Enter a 1 to enable or a 0 to disable. The default is disabled (0).

-mc Maximum cost (*maxcost*): a value that creates a priority for the connection route. The switch can select a route only if the cost does not exceed *maxcost*. The range for *maxcost* is 0–2147483647. If you do not specify *maxcost*, the connection has the highest routing priority by default. Therefore, the *maxcost* parameter lets you lower the routing priority of a connection. Note the following effects of values in the maxcost rage:

- To assign the highest priority to an SPVC based on cost (any path is acceptable), enter 2147483647. You can achieve the same result by not specifying *maxcost* at all, in which case the cost appears as a –1 in the **dspcon** output. (You cannot enter a –1 for either *maxcost* in **addcon** or to the **cnfpnni-intf** command, but display commands can show unspecified values as –1.).
- Enter a 0 for *optimal* (or least expensive) path.
- For any non-zero *maxcost*, the switch assigns a path if the total cost for all links does not exceed *maxcost*.

Although maxcost applies to an individual connection, routing costs substantially depend on a cost-per-link that you specify at every PNNI logical port in the network. The applicable PNNI command is **cnfpnni-intf**. (See the description of the *administrative weight* parameter for **cnfpnni-intf** in the chapter, “PNNI Commands.”

The cost of a route is as follows:

routing cost=sum of all costs-per-link

where:

- The cost-per-link has been specified through **cnfpnni-intf** at the egress of each logical port under PNNI control throughout the network. The impact of cost-per-link is cumulative, not just local.
- Each link has two egress points: one going to the far endpoint, and one in the return direction. The cost-per-link can differ in each direction, so the switch adds the cost-per-link in each egress instead multiplying cost by two.

The cost-per-link applies to all connections of a particular service type on a port. For example, the cost-per-link is the same for all VBR.1 connections that PNNI controls on a port, and this cost can differ from all UBR.1 connections on the same port. Alternatively, you can use **cnfpnni-intf** to make the cost-per-link the same for all service types.

To illustrate by examining a four-link route:

1. You specify a *maxcost* of 100000.
2. A potential route has four links for a total of eight egress points (four going to the endpoint and four coming back).
3. The cost-per-link at 6 ports is 5040 (the default) and 10000 at 2 ports.

The PXM45 would use the route because the resulting cost of 50240 is less than the maxcost of 100000.

Error Messages

The system can display error messages for the following reasons:

- Some of the traffic management parameters apply to specific service types (rt-VBR, for example). If you type a parameter that does not apply to a selected traffic type, the connection is rejected.
- Insufficient resources are available to accept the provisioning request.
- The type of card does not support a certain feature.
- The port cannot support SPVCs.

One of the following error messages appears if one of the preceding causes is true:

- "Port does not support requested serviceType"
- "lscr/lmcr not allowed to exceed lpcr (dcmp)"
- "rscr not allowed to exceed rpcr"
- "lpcr must be defined for cbr serviceType"
- "rpcr must be defined for cbr serviceType"
- "lpcr and lscr must be defined for vbr serviceType"
- "rpcr and rscr must be defined for vbr serviceType"
- "lpcr must be defined for abr/ubr serviceType"
- "rpcr must be defined for abr/ubr serviceType"
- "Requested rcdv is too low"
- "Requested rctd is too low"
- "Requested max cell loss ratio (clr) is too high"
- "Requested cell rate (lscr/lpcr) is too high"
- "Requested cell rate (rscr/rpcr) is too high"

Related Commands

cnfcon, **cnfabr**, **delcon**, **dspcon**, **dspcons**, **dncon**, **upcon**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Add the slave end of a VCC on logical port 1 with VPI=10, VCI=40, CBR service type. Note that the system returns the slave end connection identifier in the hexadecimal NSAP format with the VPI.VCI at the end. When you add the master endpoint of the connection, type –slave followed by this connection identifier. You can do a copy and paste rather than typing the whole string.

```
MGX8850.AXSM.a >addcon 1 10 40 1 s
slave endpoint added successfully
slave endpoint id: 00000E1000001C008051B730FFFFFF010B180100.10.40
```

In the following two examples, the connection works with default values of PCR, SCT, MCR taken from the SCT. Defaults applied for the connection can be viewed by using the **dspcon** command.

```
MGX8850.1.11.AXSM.a > addcon 1 10 40 1 s
slave endpoint added successfully
slave endpoint id : 00000E1000001C008051B730FFFFFF010B180100.10.40

MGX8850.1.11.AXSM.a > addcon 1 10 50 1 m -slave
00000E1000001C008051B730FFFFFF010B180100.10.40
master endpoint added successfully
master endpoint id : 00000E1000001C008051B730FFFFFF010B180100.10.50
```

In the following two examples, the connection works with default values of SCR, MCR derived from the PCR value specified using **lpcr** and **rpcr** keywords. Defaults applied for the connection can be viewed by using the **dspchan** command.

```
MGX8850.1.11.AXSM.a > addcon 1 10 40 1 s
slave endpoint added successfully
slave endpoint id : 00000E1000001C008051B730FFFFFF010B180100.10.40

MGX8850.1.11.AXSM.a > addcon 1 10 50 1 m -slave
00000E1000001C008051B730FFFFFF010B180100.10.40 -lpcr 1000 -rpcr 1000
master endpoint added successfully
master endpoint id : 00000E1000001C008051B730FFFFFF010B180100.10.50
```

clrpncon

Clear Connection

Delete a call—either all SVCs or a specific SVC on a port. (If you attempt to clear an SPVC with this command, the switch deletes the connection but then automatically attempts to reroute it. For an SPVC or SPVP, use **delcon** so you can delete the endpoints.)

Cards on Which This Command Runs

PXM45

Syntax

```
clrpncon <portid>
{all | vpi}
[vci]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
all | vpi	Specifies either all VPIs on the port or a specific VPI. Possible values are either the string “all” or a VPI in the range 0–4095.
vci	VCI of a specific SVC to clear. If you are clearing a virtual path connection (VPC), do not enter a VCI.

Related Commands

dsppncon, dspncons

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

First, list the connections on the port to identify the specific connection to delete. For this example, use **clrpncon** to release the connection on port 1.2 with the VPI/VCI or 1 100. This connection is the first in the display output. Thereafter, use **dsppncons** to check the results.

```
Geneva.7.PXM.a > dsppncons

Port       VPI  VCI       CallRef X-Port       VPI  VCI   CallRef  Type OAM-Type
1:1.2:2     1   100       33       1:1.2:2      1   101      32    PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     1   101       32       1:1.2:2      1   100       33   PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     2   200       34       1:1.6:6      0    49   8388609  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.6:6     0    49  8388609       1:1.2:2      2   200        34  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00

Geneva.7.PXM.a > clrpncon 1.2 1 100

Geneva.7.PXM.a > dsppncons
Port       VPI  VCI       CallRef X-Port       VPI  VCI   CallRef  Type OAM-Type
1:1.2:2     2   200       34       1:1.6:6      0    49   8388609  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.6:6     0    49  8388609       1:1.2:2      2   200        34  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00

Geneva.7.PXM.a >
```

clrpnconstats

Call Control Operations

Clears existing call statistics for one logical port or all logical ports.

Cards on Which This Command Runs

PXM45

Syntax

`clrpnconstats` [*portid*]

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.

Related Commands

dsppnconstats

Attributes

Log: log	State: active	Privilege: SUPER_GP

cnfabr

Configures the ABR-specific parameters for an existing SPVC. The connection must be of service type ABR (in the **addcon** command, *service type*=10).

Cards on Which This Command Runs

AXSM

Syntax

cnfabr <*ifNum*> <*vpi* <*vci*>
[**-icr** <*Initial cell rate*>]
[**-adtf** <*ACR decr. factor*>]
[**-rdf** <*Rate decr. factor*>]
[**-rif** <*Rate incr. factor*>]
[**-nrm** <*Cells per fwd RM*>]
[**-trm** <*Time between fwd RMs*>]
[**-cdf** <*cutoff decrease factor*>]
[**-frtt** <*fix round trip delay*>]
[**-tbe** <*transient buffer exposure*>]
[**-intvsvd** <*internal vsvd config*>]
[**-extvsvd** <*external vsvd config*>]

Syntax Description

ifNum	The port number of the connection to configure.
vpi	The VPI range for a UNI port endpoint is 0–255. The VPI range for an NNI or VNNI port endpoint is 0–4095.
vci	The VCI range for a UNI port endpoint is 1–4095. The VCI range for a NNI port endpoint is 32–65535. For MPLS, the recommended minimum VCI is 35.
-icr	Keyword that specifies the Initial Cell Rate (ICR) in cells per second. This is the rate at which the source should begin transmitting, and is also the rate at which the source should resume transmitting after an idle period. The range is 0–4294967295 cells per second.
-adtf	Keyword that specifies the ACR Decrease Time Factor (ADTF). This is the time permitted to decrease the cell rate from the RM-cell rate to the Allowed Cell Rate (ACR) for normal traffic. The range is 1–1023 milliseconds.
-rdf	Keyword that specifies the Rate Decrease Factor (RDF). This is the factor by which to decrease the Allowed Cell Rate (ACR). *RDF* is a power of 2 in the range 1/32768 to 1.
-rif	Keyword that specifies the Rate Increase Factor (RIF). This is the factor by which to increase the Allowed Cell Rate (ACR). *RIF* is a power of 2 in the range 1/32768 to 1.

-nrm	Keyword that specifies the maximum number of cells that the source can send for each forward RM-cell. *Nrm* is a power of 2 in the range 2–256.
-trm	Keyword that specifies the maximum number of milliseconds for one RM-cell to travel from source to endpoint. The range is 100 x 2^{-7} to 100 x 2^{0} milliseconds.
-cdf	Keyword that specifies the Cutoff Decrease Factor (CDF). This value controls the decrease in Allowed Cell Rate (ACR) associated with Missing RM-cell count (CRM). *CDF* can be either or the following: • Zero • A power of 2 in the range 1/64 to 1 CRM limits the number of forward RM-cells that may be sent in the absence of received backward RM-cells. CRM is an integer. Its size is implementation specific.
-frtt	Keyword that specifies the Fixed Round-Trip Time (FRTT). This is the sum of the fixed delays plus the propagation delays from the source to the destination and back. The range is 0– 16.7 seconds.
-tbe	Keyword that specifies the Transient Buffer Exposure (TBE). This is the negotiated number of cells that the network would like to limit the source to sending during startup periods, before the first RM-cell returns. The range is 0–16,777,215 cells.
-intvsvd	Keyword that specifies the internal virtual source / virtual destination (VS/VD). • 1=Off • 2=On • 3=Unspecified
-extvsvd	Keyword that specifies the external virtual source / virtual destination (VS/VD). • 1=Off • 2=On • 3=Unspecified

Related Commands

addcon, cnfabrtparmdft, dspabrtparmdft

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.10.AXSM.a > cnfabr 1 77 777 -mcr 100
 THE SG NUM is: 0.
Configuration successful
```

cnfabrtparmdft

Configure ABR Traffic Parameter Defaults

Configures the default ABR traffic parameter(s), used for SPVC, per port.

Note Currently, this command is available but has no effect.

When a user requests an ABR connection, the controller adds the default ABR traffic parameters before the connection is committed. The user can change the ABR traffic parameters, per connection, using the **cnfabr** command available on the AXSM.

The default ABR traffic parameters are used in the SETUP message at the source when an SPVC for ABR service category is set up.

Cards on Which This Command Runs

PXM45

Syntax

cnfabrtparmdft <*portid*>
[**-rif** *RIF-value*]
[**-rdf** *RDF-value*]
[**-tbe** *TBE-value*]
[**-nrm** *NRM-value*]
[**-trm** *TRM-value*]
[**-adtf** *ADTF-value*]
[**-cdf** *CDF-value*]
[**-fsd** *FSD-value*]

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
-rif	Keyword that specifies the Rate Increase Factor (RIF). This is the factor by which to increase the Allowed Cell Rate (ACR). *RIF* is a power of 2 in the range 1/32768 to 1.
-rdf	Keyword that specifies the Rate Decrease Factor (RDF). This is the factor by which to decrease the Allowed Cell Rate (ACR). *RDF* is a power of 2 in the range 1/32768 to 1.
-tbe	Keyword that specifies the Transient Buffer Exposure (TBE). This is the negotiated number of cells that the network would like to limit the source to sending during startup periods, before the first RM-cell returns. The range is 0–16,777,215 cells.
-nrm	Keyword that specifies the maximum number of cells that the source can send for each forward RM-cell. *Nrm* is a power of 2 in the range 2–256.

-trm Keyword that specifies the maximum number of milliseconds for one RM-cell to travel from source to endpoint. The range is 100 x 2^{-7} to 100 x 2^{0} milliseconds.

-adtf Keyword that specifies the ACR Decrease Time Factor (ADTF). This is the time permitted to decrease the cell rate from the RM-cell rate to the Allowed Cell Rate (ACR) for normal traffic. The range is 1 to 1023 milliseconds.

-cdf Keyword that specifies the Cutoff Decrease Factor (CDF). This controls the decrease in Allowed Cell Rate (ACR) associated with Missing RM-cell count (CRM). *CDF* can be either or the following:

- Zero
- A power of 2 in the range 1/64 to 1

CRM limits the number of forward RM-cells that may be sent in the absence of received backward RM-cells. CRM is an integer. Its size is implementation-specific.

-fsd Keyword that specifies the Fixed-source-delay (default = 0).

Related Commands

addcon, cnfabr, dspabrtparmdft

Attributes

Log: no log	State: active	Privilege: GROUP1

cnfcdvtdft

Configure Cell Delay Variation Tolerance Default

For all connections of a particular service type on a PNNI logical port, **cnfcdvtdft** configures the default number of microseconds for the cell delay variation tolerance (CDVT). The direction is ingress. The new configuration applies to new incoming calls but not existing calls. You can execute **cnfcdvtdft** whether the port is in the provisioning state (prior to **addport** on the service module) or administratively up.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfcdvtdft <portid> <service_category>
[microseconds]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
<*service_category*>	Service type: **cbr**, **rtvbr**, **nrtvbr**, **ubr**, or **abr**.
micro seconds	The number of microseconds for CDVT. Range: 0–2147483647 Default = 250,000

Related Commands

dspcdvtdft

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Specify a CDVT of 125000 microseconds for ABR connections on port 4:1.1:11. Check the results by executing **dspcdvtdft** for the port.

```
Geneva.7.PXM.a >   cnfcdvtdft 4:1.1:11 abr 125000

Geneva.7.PXM.a > dspcdvtdft 4:1.1:11

              cbr:     rt-vbr:    nrt-vbr:        ubr:        abr:
CDVT:       250000      250000      250000      250000     125000

Geneva.7.PXM.a >
```

cnfcon

Configure Connection

Modifies the bandwidth, policing, and routing parameters of an existing endpoint. This command applies to only an SPVC or SPVP. For ABR-specific parameters, use the **cnfabr** command.

The command parameters consist of:

- A logical port, VPI, and VCI to identify the connection
- Bandwidth parameters for the local (master) end then the remote (slave) end
- Policing parameters for the connection as a whole

After you specify the mandatory connection identifier, all other parameters are optional.

Card(s) on Which the Command Executes

AXSM

Usage Guidelines for cnfcon

The following sections discuss the application of certain **cnfcon** parameters.

Note On DAX connections, using **cnfcon** at the slave end has no effect. For DAX connections, use **cnfcon** at the master end only, and the parameters will take effect on the controller as well.

Traffic Parameters

Traffic parameters such as PCR, SCR, MBS are entered at both the master and slave endpoints for both the forward and reverse directions. Be sure that the value entered as “local” on one end is equal to the value entered as “remote” on the other end. For example, the *lpcr* on the slave endpoint should be same as the *rpcr* on the master endpoint and vice versa when you provision the connection at the other end. If you modify traffic parameters after creating an SPVC, you just modify them at either the master endpoint or the slave endpoint.

Traffic parameters such as CDV, CTD are entered at both the master and slave endpoints for both the forward and reverse directions. However, the values of these parameters entered at the slave end are ignored during call setup. Therefore, you can specify the *lcdv*, *rcdv*, *lctd* and *rctd* options at the master end only.

Routing Parameters

The routing parameter *maxcost* (specified using -**mc** option) need to be entered at the master endpoint only. The values of this parameters entered at the slave end is ignored and not considered during call setup.

Frame Discard

For the parameter *frame discard* (specified using -**frame** option), you need to enter it at only the master endpoint. This parameters has no significance at the slave end.

For the MGX 8850 2.0 release, if you try to enable frame discard at the slave end point you will not get an error message. Nothing will happen, and frame discard will not take effect. In future releases, an error message will be displayed if you try to enable frame discard at the slave end point.

Local-Only Parameters

The parameters CDVT, stats enable, cc enable (specified using **-cdvt**, **-stat**, **-cc**) are significant only at the endpoint where you enter them. Therefore, they can be different at each end of the connection.

AXSM Syntax

```
cnfcon
<ifNum>
<vpi>
<vci>
[-lpcr <local to remote PCR>]
[-rpcr <remote to local PCR>]
[-lscr <local to remote SCR>]
[-rscr <remote to local SCR>]
[-lmbs <local to remote MBS>]
[-rmbs <remote to local MBS>]
[-lcdv <local to remote maxCDV>]
[-rcdv <remote to local maxCDV>]
[-lctd <local to remote maxCTD>]
[-rctd <remote to local maxCTD>]
[-lmcr <local to remote MCR>]
[-rmcr <remote to local MCR>]
[-cdvt <local CDVT>]
[-cc <OAM CC Cnfg>]
[-stat <Stats Cnfg>]
[-frame <frame discard>]
[-mc <Max Cost>]
[-segep <OAM segment endpoint>]
```

AXSM Syntax Description

ifNum — Logical port number. On the AXSM, the range is 1–60.

vpi — Virtual path identifier in the range 0–255 (UNI) or 0–4095 (NNI or VNNI).

vci — Virtual connection identifier (VCI):

- For a VCC on a UNI, the range is 1–4095. On an NNI or VNNI, the VCI range is 32–65535. For MPLS, the recommended minimum VCI is 35.
- For a VPC, the *vci* is 0.

- lpcr — Keyword that specifies the Peak Cell Rate (PCR) from a local endpoint to a remote endpoint (7–5651328 cells per second). Peak Cell Rate is the maximum cell rate for the connection at any time.

- rpcr — Keyword that specifies the Peak Cell Rate (PCR) from a remote endpoint to a local endpoint (7–5651328 cells per second). Peak Cell Rate is the maximum cell rate for the connection at any time.

- lscr — Keyword that specifies the Sustained Cell Rate (SCR) from a local endpoint to a remote endpoint (7–5651328 cells per second). Sustained Cell Rate is the maximum cell rate that a connection can sustain for long periods.

- rscr	Keyword that specifies the Sustained Cell Rate (SCR) from a remote endpoint to a local endpoint (7–5651328 cells per second). Sustained Cell Rate is the maximum cell rate that a connection can sustain for long periods.
- lmbs	Keyword that specifies the Maximum Burst Rate (MBS) from a local endpoint to a remote endpoint (1–5000000 cells). Maximum Burst Size is the maximum number of cells that can burst at the PCR and still be compliant.
- rmbs	Keyword that specifies the Maximum Burst Rate (MBS) from a remote endpoint to a local endpoint (1–5000000 cells). Maximum Burst Size is the maximum number of cells that can burst at the PCR and still be compliant.
- cdvt	Keyword that specifies the Cell Delay Variation Tolerance (CDVT) from a local endpoint to a remote endpoint (1–5000000 microseconds). Cell Delay Variation Tolerance controls the time scale over which the PCR is policed. Note that no remote CDVT is necessary.
- rcdvt	Keyword that specifies the Cell Delay Variation Tolerance (CDVT) from a remote endpoint to a local endpoint (1–5000000 microseconds). Cell Delay Variation Tolerance controls the time scale over which the PCR is policed.
- lcdv	Keyword that specifies the Cell Delay Variation (CDV) from a local endpoint to a remote endpoint (1–16777215 microseconds). Cell Delay Variation is the peak to peak cell delay variation expressed in microseconds.
- rcdv	Keyword that specifies the Cell Delay Variation (CDV) from a remote endpoint to a local endpoint (1–16777215 microseconds). Cell Delay Variation is the peak to peak cell delay variation expressed in microseconds.
- cc	(Optional) Keyword that sets the OAM CC: Enter 1 to enable or 0 to disable. The default is 0.
- stat	(Optional) Keyword that sets the statistics collection: Enter 1 to enable or 0 to disable. The default is 0. The Cisco WAN Manager tool collects statistics for a connection if you enable it here. Statistics collection is disabled for all connections by default. Statistics collection has an impact (which may not be significant) on the real-time response, especially for SVCs (which can be affected even though you do not add SVCs). Therefore, you should enable statistics collection for only that subset of connections that really warrants such a feature.
- frame	Keyword that sets frame discard: Enter a 1 to enable or a 0 to disable. The default is 0.

- mc	(Optional) Keyword that specifies the maximum cost (*maxcost*): a value that creates a routing priority. If you do not specify this optional parameter, the connection defaults to having the highest routing priority. Therefore, the *maxcost* parameter lets you lower the priority of a connection—but only in regards to finding a route for it. The range for *maxcost* is 0–2147483647. The PXM45 does not use a particular route if the *cost* for the route exceeds the *maxcost*. The *cost* of a route depends on a *cost-per-link* specified through the **cnfpnni-intf** command. The cost-per-link applies to at the egress a port for all connections of a particular service type. For example, the cost-per-link is the same for all VBR.1 connections that PNNI controls on a given port, but this cost can differ from all UBR.1 connections on the same port. For a route under consideration, the cost is the sum of all the cost-per-links at each egress in the forward and backward directions along the entire route. In a one-link route, for example, the cost is the sum of the cost-per-links at two ports. To illustrate further with a four-link route: 1. You specify a *maxcost* of 100000. 2. The route under consideration by the switching fabric has four links for a total of eight egress points. 3. The cost-per-link at six ports is 5040 (the default in **cnfpnni-intf**), and the cost per link at two ports is 10000. The node would use the route because the resulting cost of 50240 is less than the maxcost of 100000. Note that you can specify no limit for a service type by entering a cost-per-link of -1 through **cnfpnni-intf**. A -1 cost-per-link makes *maxcost* meaningless.
-segep	OAM segment endpoint: Enter a 1 to enable or a 0 to disable.

Related Commands

addcon, **delcon**, **dspcon**, **dspcons**, **dspconstats**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.11.AXSM.a > cnfcon 1 10 40 -cc 1
Configuration successful
```

cnfconsegep

Configure Connection Segment Endpoint

Configures a connection as a segment endpoint. When both VPI and VCI are present, the segment endpoint is an F5 flow endpoint (for VCCs). When the optional VCI is not present, the segment endpoint is an F4 flow endpoint (for VPCs). Use this command only for established calls.

Note Before executing this command, be sure continuity checking is de-activated. If you leave it on, continuity check failure occurs for the connection. Use **cnfconsegep** in conjunction with **conntrace** for fault isolation.

Command(s) on Which Command Executes

PXM45

Syntax

```
cnfconsegep <portid>
< vpi >
[vci]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
vpi	VPI of the connection.
vci	VCI of the connection.

Related Commands

cnfoamsegep, dspoamsegep, delconsegep, dspconsegep

Attributes

Log: log	State: active	Privilege: GROUP1

cnfmbsdft

Configure Maximum Burst Size Default

Configures the default maximum burst size (MBS) for SPVCs on a port. The applicable service types are real-time and non real-time variable bit rate (rt-VBR and nrt-VBR).

The most likely connection type for which you would use **cnfmbsdft** is SVC. You can also rely on the value set with this command as a default for SPVCs if you do not specify an MBS through **addcon** for each SPVC of service type VBR.

The new configuration applies to new incoming calls, not existing calls. You can use **cnfmbsdft** whether the port is active or in the provisioning state.

Cards on Which This Command Runs

PXM45

Syntax

```
cnfmbsdft <portid> <service_category>
[num-of-cell]:
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
service_category	ATM 4.0 service category—either rtvbr or nrtvbr.
num-of-cell	The units of measure for MBS are cells. Range: 0–2147483647 cells Default: set by the platform to 1024 cells.

Related Commands

dspmbsdft

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

Configure a default MBS of 10000 cells for nrt-VBR.

```
cnfmbsdft 11:2.1:1 nrtvbr 10000
```

cnfoamsegep

Configure OAM Segment Endpoint

Define the port as a segment endpoint for F4 and F5 operations administration and maintenance (OAM) cells. This command does not take effective for existing connections, and only does for newly established calls. This command can be used regardless of the state of the port.

Command(s) on Which Command Executes

PXM45

Syntax

```
cnfoamsegep <portid>
[{yes | no}]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
yes	The port is configured as a segment endpoint and is a segment endpoint for all connections on this port.
no	The port is not a segment endpoint. Default = no

Related Commands

dspoamsegep, cnfconsegep, delconsegep

Attributes

Log: log	State: active	Privilege: GROUP1

delcon

Delete Connection

Use the **delcon** command to delete an SPVC or SPVP. Delete the connection at both ends—first at the master end, then at the slave.

Cards on Which This Command Runs

AXSM

Syntax

`delcon` *`<ifnum> <vpi> <vci>`*

Syntax Description

ifnum	Logical port number. On the AXSM, the range is 1–60.
vpi	Virtual path identifier in the range 0–255 (UNI) or 0–4095 (NNI or VNNI).
vci	Virtual connection identifier (VCI): • For a VCC on a UNI, the range is 1–4095. On an NNI or VNNI, the VCI range is 32–65535. For MPLS, the recommended minimum VCI is 35. • For a VPC, the *vci* is 0.

Related Commands

dspcon, **addcon**, **cnfcon**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

```
MGX8850.1.3.AXSM.a > delcon 1 10 40
Deletion successful
```

delcons

Delete Connections

Delete a range of ATM connections.

Caution Before entering this command, you need to know exactly how many connections you need to delete. This command starts deleting connection endpoints in lexicographic order (**dspcons** displays in the same order). Incorrect usage of this command could result in deletion of more endpoints than necessary, and repairing such damage could be very costly.

Cards on Which This Command Runs

AXSM

Syntax

```
delcons <ifNum> <vpi> <vci>
[-num <num. conns to del>]
[-verbose < 1 | 0 >]
```

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	For a UNI, the range is 0–255. For an NNI, the range is 0–4095.
vci	For a VCC, the range is 32–65535. For a VPC, the only value is 0.
-num	(Optional) Keyword that specifies the number of connections to delete.
-verbose	(Optional) Keyword that enables (1) or disables (0) verbose mode. In verbose mode, the screen displays the connection identifier of each connection immediately after it is deleted.

Related Commands

None

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

delconsegep

Delete Connection Segment Endpoint

Deletes a segment endpoint on a connection. When both VPI and VCI are present, the segment endpoint is an F5 flow endpoint (for VCCs). When the optional VCI is not present, the segment endpoint is an F4 flow endpoint (for VPCs).

Note The **delconsegep** command works for SVCs only.

Cards on Which This Command Runs

PXM45

Syntax

```
delconsegep <portid>
vpi
[vci]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
vpi	VPI of the connection.
vci	VCI of the connection.

Related Commands

cnfoamsegep, dspoamsegep, cnfconsegep, dspconsegep

Attributes

Log: log	State: active	Privilege: GROUP1

dncon

Down Connection

Administratively deactivates (or “downs”) so you can modify or troubleshoot the network. This operation applies to only SPVCs. To reactivate the connection, use **upcon**.

Cards on Which This Command Runs

AXSM

Syntax

dncon
<*ifNum*>
<*vpi*>
<*vci*>

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	Virtual path identifier. On a UNI, the range is 0–255. On an NNI, the range is 0–4095.
vci	For a virtual connection (VCC) on a UNI, the range is 1–4095. On an NNI or VNNI, the VCI range is 32–65535. For MPLS, the recommended minimum VCI is 35. For a virtual path connection (VPC), the VCI is always 0.

Related Commands

upcon

Attributes

Log: log	State: active	Privilege: GROUP1

dspabrtparmdft

Display ABR Parameter Defaults

Displays the default ABR parameters for a logical port under PNNI.

Cards on Which This Command Runs

PXM45

Syntax

dspabrtparmdft <*portid*>

Syntax Description

portid has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].

Related Commands

addcon, cnfabr, cnfabrtparmdft

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the default ABR parameters on portid 1:1.1:1.

```
Default ABR Traffic Parameters For: 1:1.1:1
-----------------------------------
RIF: 7 (= 1/512)
RDF: 4 (= 1/4096)
TBE: 1048320 (Cells)
NRM: 5 (= 32 Cells)
TRM: 8 (= 100 misc.)
ADTF: 50 (= 0.50 Sec)
CDF: 7 (= 1/2)
FSD: 0 (microSec)
```

dspcdvtdft

Display CDVT Default

Displays the default CDVT for the port.

Cards on Which This Command Runs

PXM45

Syntax

`dspcdvtdft` <*portid*>

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.

Related Commands

dspcdvtdft, cnfcdvtdft

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the CDVT defaults for port 11:1.1:11.

```
Geneva.7.PXM.a > dspcdvtdft 11:1.1:11

              cbr:     rt-vbr:    nrt-vbr:        ubr:        abr:
CDVT:       250000      250000      250000      250000      250000
```

dspcon

Display Connection

Display information about an SPVC. The contents of the display on the AXSM and the PXM45 differ slightly. On both cards, the **dspcon** output appears in sections to make the information easier to sort.

Most of the information in the **dspcon** output comes from **addcon** execution. See the **addcon** description for more information. Also, executing **cnfpnni-intf** can affect the **dspcon** output.

Display Connection on the PXM45

On the PXM45, **dspcon** shows the following connection identifiers:

- NSAP address, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type and compliance (for example, UBR for service type and UBR.1 for ATM Forum compliance).
- Bearer class (relates to voice traffic and is reserved for future use).
- Whether continuity checking or frame discard are enabled (see **addcon** description).
- Cause of the last failure. This field can also show that no errors have occurred since the connection was first added by displaying "SPVC Established." If a failure occurred, the Attempts field shows the number of times the system attempted to re-establish service. If no failures have occurred, the Attempts field contains a 0.
- L-Util and R-Util are the local and remote percent of utilization assigned to the connection. Currently, the default of 100% is the only value.
- Cost values for the connection's route: the two fields in this category are Max Cost and Routing Cost. The Max Cost is a cost-per-link configured for a service type (such as UBR) through the **cnfpnni-intf** command. When you add the SPVC through **addcon**, you can specify a maximum routing cost through the maximum cost (maxcost) parameter. The maxcost represents the maximum cost for an individual connection. The system uses the cost-per-link for the service type and the maxcost for the connection to determine whether a route costs too much. After the system creates a route, the total number of links yields the Routing Cost.

  The default cost-per-link is 5040, so if a particular service type uses the default and a route consists of 4 links, the Routing Cost is 20160. If the **dspspvc** display shows that Max Cost is –1, no limit was specified through **cnfpnni-intf**, and the resulting Routing Cost is 0.
- Broadcast type: point-to-point or multicast.

The Traffic Parameters section shows the standard parameters PCR, SCR, and CDV in the receive and transmit directions.

Display Connection on the AXSM

On the AXSM, **dspcon** shows the following connection identifiers:

- NSAP address, logical port, VPI/VCI, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type (for example, ABR).
- A number indicating the controller. For example, 2 refers to PNNI. The **addcontroller** command specifies the controller.
- The administrative state is either up or down. This state results from **addcon** or **dncon**/**upcon**. Note that, after you down a connection with at the connection master endpoint, the **dspcon** command shows the connection as "down" when you execute it at the master endpoint and "failed" when you execute it at the slave endpoint. (See also **dncon** description).
- The operational state is either OK or failed. The operational state can apply to a connection regardless of the administrative state.

The traffic management parameters consist of:

- Local and remote UPC parameters of PCR, MBS, CTD, CDVT, and so on. A –1 in a field means that the parameter was not specified. The characters "N/A" indicate that the parameter does not apply to the service type.

These other fields also pertain to connection integrity:

- OAM connectivity check enable or disable.
- Loopback test enable/disable and loopback type.
- Round trip delay in microseconds. This field is non-zero only if you previously executed **tstdelay**.

The **dspcon** command requires a unique connection identifier. If you do not have the information to identify a connection, execute **dspcons**. On the AXSM, **dspcons** identifies all the connections on the AXSM. On the PXM45, **dspcons** identifies all the connections on the node. (See **dspcons** description).

Cards on Which This Command Runs

AXSM, PXM45

Syntax

On the AXSM:

dspcon *<ifNum>* *<vpi>* *<vci>*

On the PXM45:

dspcon *<portid>* *<vpi>* *<vci>*

Syntax Description

ifNum	(AXSM) Logical interface (port) number. For AXSM, the range is 1–60.
portid	(PXM45) The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
vpi	VPI number. At the UNI, the range is 0–255. At the NNI, the range is 0–4095.
vci	VCI number. For a VCC, the range is 32–65535. For a VPC, the VCI is 0.

Related Commands

addcon, **dspcons**, **cnfcon**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display connection 5 31 63000 on the current AXSM.

```
MGX8850.1.AXSM.a > dspcon 5 31 63000
--------------------------------------------------------------------------

Local   :               NSAP  Address               port     vpi    vci

(M)       47009181000000000107BE92F3F00000101180500  1.01.05 31 63000
Remote  :               NSAP  Address               port     vpi    vci

(S)       47009181000000000107BE92F3F00000101180500  1.01.05   3201 100
--------------------------------------------------------------------------

Conn. Type    :     VCC                          Admn Status  : ADMN-UP
Service Type  :    cbr1                          Oper Status  : FAIL
Controller    :       2
--------------------------------------------------------------------------

Local PCR     :   10000                          Remote PCR   : 1000
Local SCR     :     N/A                          Remote SCR   : N/A
Local CDV     :      -1                          Remote CDV   : -1
Local CTD     :      -1                          Remote CTD   : -1
Local MBS     :     N/A                          Remote MBS   : N/A
Local CDVT    :      -1                          Remote CDVT  : -1
Admin weight  :      -1                          Frame discard: N
--------------------------------------------------------------------------

OAM CC Config :DISABLED                          Statistics   : DISABLED
--------------------------------------------------------------------------

Loopback Type :No  Lpbk | Dir:N/A     | Status: No Lpbk | RTD: 0 us
```

On the CLI of the PXM45, display connection 20 100 on 11:1.1:2.

```
Unknown.7.PXM.a > dspcon 11:1.1:2 20 100
Port                   Vpi Vci                        Owner      State
-------------------------------------------------------------------------
Local  11:1.1:2        20.100                         MASTER     FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00
Remote 11:1.1:2        10.100                         SLAVE      FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: CBR         Conformance: CBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  50             Rx PCR:  50
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

Display information for vpi/vci 10 100 on port ID 1:1.1:1. In this case, port ID and remote and local NSAP addresses are the same, so the connection is a DAXCON. Also, the Max Cost is –1. The Max Cost of -1 means no cost-per-link was specified for UBR service, and therefore the Routing Cost is 0.

```
node19.8.PXM.a > dspcon 1:1.1:1 10 100
Port                   Vpi Vci                        Owner      State
-------------------------------------------------------------------------
Local  1:1.1:1         10.100                         SLAVE      OK
       Address: 47.00918100000000001a53c82d.000001011801.00
Remote 1:1.1:1         11.101                         MASTER     OK
       Address: 47.00918100000000001a53c82d.000001011801.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: UBR         Conformance: UBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  14             Rx PCR:  14
Tx SCR:  3              Rx SCR:  3
Tx MBS:  1              Rx MBS:  1
Tx CDVT: -1             Rx CDVT: -1
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

dspconinfo

Display Connection Information—display summaries of connection configuration (SPVCs or SPVPs only)

The **dspconinfo** runs on the PXM45 and lists the total number SPVCs on each PNNI port on the node. For each port, the display shows:

- Number of active connections
- Number of failed connections
- Number of down connections
- Total number of connections

Cards on Which This Command Runs

PXM45

Syntax

dspconinfo

Syntax Description

This command takes no parameters.

Related Commands

none

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the SPVC summaries for the current node. This example shows all connections as failed.

```
popeye10.7.PXM.a > dspconinfo
Local Port     #Active   #Failed      #Down     #Total
------------------------------------------------------------
2:2.2:1             1         0          0          1
3:1.1:1             1         0          0          1
```

dspcons

Display Connections—display basic information for all connections.

The default usage of **dspcons** uses no parameters and causes all available information for the connections to appear. To narrow the scope of the output, use one or more optional parameters.

The **dspcons** command runs on the CLI of either the AXSM or the PXM45. The set of optional parameters and the output are different on these cards. (See Syntax Description for the card-specific parameters.) On the AXSM, the columns at the head of the information fields are:

record	A number for the connection with internal application only. It resides in the database on the AXSM and is not affected by user input. The system creates this number when you create the connection. The Cisco WAN Manager application uses this number.
Identifier	Identifies the connection in the format *port vpi vci*.
Type	Shows whether the connection is a VCC or a VPC.
SrvcType	The service type—VBR, and so on. (See **addcon** description).
M/S	Indicates whether the endpoint specified by *Identifier* is the master or slave.
Upld	The hexadecimal Upload number is an encoded timestamp the Cisco WAN Manager application uses to determine when a connection was created or modified. In the CLI context, this field has little meaning.
Adm	The administrative state of the connection. If the connection is down, it may have resulted from the **dncon** command.
Alarm	Shows the alarm status of the connection.

When you execute **dspcons** on the CLI of the PXM45, the output shows:

Local and Remote Port ID	The display contains a column for the local port ID and a column for the remote port ID. The port ID has the format that the network controller utilizes: [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]
Local and Remote VPI.VCI	The VPI and VCI at the local and remote ends of the connection.
State	The State column shows whether the connection is OK, down (by the **dncon** command), failed, or has an alarm indication signal (AIS) or *abit* errors.
Owner	Whether the endpoint is master or slave.
Local and Remote NSAP	An NSAP address for each end of the connection.

Cards on Which This Command Runs

PXM45, AXSM

PXM45 Syntax

```
dspcons
[-port portid]
[-vpi starting-vpi]
[-vci starting vci]
[-state {fail|ais|abit|ok|down}]
[-owner {master|slave}]
[-sc service class]
```

PXM45 Syntax Description

-**port**	The port identifier (*portid*) in the format that the network controller utilizes: [*shelf.*]*slot*[:*subslot*].*port*[:*subport*] Currently, the value for *shelf* is always 0 and therefore is not necessary.
-**vpi**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**vci**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**state**	A specific connection state. The display shows only the connections with the state you specify. Note that on the PXM45, you must spell out the entire state keyword. The keywords for specifying a state are **failed**—only failed connections **ais**—connections with alarm indication signal (AIS) set **abit**—connections on which an A-bit error has occurred **ok**—connections with no problems **down**—connections that are administratively down because a user has executed **dncon** to down the connection
-**owner**	Specifies connections where the local endpoint is either master or slave.
-sc	Specifies the service class: ABR, VBR, CBR, UBR.

AXSM Syntax

```
dspcons
[-conn <conn id>]
[-filt <filter options>]
[-if <intf no>]
[-vpi <vpi filter>]
[-vci <vci filter>]
```

AXSM Syntax Description

-conn	The connection ID (*conn ID*) of the connection to begin the display. The format of *conn ID* is: *ifNum.vpi.vci* The logical port number. The range for AXSM is 1–60. The VPI has the range 0–4095. The VCI has the range 32–65535.
-filt	Unlike on the PXM45, you do not use keywords for this parameter on the AXSM. You enter only a number on the AXSM CLI to indicate the state. The display criteria are: 1 ingr—for errors in the ingress direction 2 egr—for errors in the egress direction 3 condn—for connections where the switch has conditioned the connection 4 iffail—for connection on a failed logical interface 5 ccfail 6 mis 7 abit—for connections where an A-bit error has occurred
-if	A particular logical interface for connection display.
-vpi	The VPI of all the connections that you would like to display.
-vci	The VCI of all the connections that you would like to display.

Related Commands

dspcon, **addcon**, **cnfcon**, **delcon**, **dncon**, **upcon**

PXM45 Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active, standby	Privilege: GROUP1

PXM45 Example

Display all connections by entering **dspcons** on the CLI of the PXM45.

```
MGX8850.7.PXM.a > dspcons

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
3:1.1:1            20 0      6:1.1:1            20 0       OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.000001031801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
5:1.1:1           100 100    5:1.1:1           100 200     OK         SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
5:1.1:1           100 200    5:1.1:1           100 100     OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
6:1.1:1            20 0      3:1.1:1            20 0       OK         SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001031801.00
6:1.1:1           100 100    6:1.1:1           100 200     OK         SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           100 200    6:1.1:1           100 100     OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK         SLAVE

6:1.1:1           100 200    6:1.1:1           100 100     OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK         SLAVE

Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061803.00
6:2.1:3           200 200    6:1.1:1           200 100     OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061803.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
9:1.3:3            10 100    Routed              0 0       FAIL       SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001091803.00
Remote Addr: 00.000000000000000000000000.000000000000.00
11:1.1:2           10 100    11:1.1:2           20 100     OK         SLAVE
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
11:1.1:2           20 100    11:1.1:2           10 100     OK         MASTER
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00
```

AXSM Example

Display all connections on the current AXSM. In this example, only one connection exists. Master and slave endpoints are shown.

```
GN.6.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld     Admn   Alarm
------    ----------   ----   --------   ---    ----     ----   -----
    0  01.0010.00100   VCC        cbr1   S   010c7953    UP      none
    1  04.0020.00100   VCC        cbr1   M   010c7964    UP      none
```

dspconsegep

Display Connection Segment Endpoint

Displays OAM segment endpoint for a connection endpoint. When both VPI and VCI are present, the segment endpoint is an F5 flow endpoint (for VCCs). When the optional VCI is not present, the segment endpoint is an F4 flow endpoint (for VPCs). This command is used only for established calls.

Note The **delconsegep** command works for SVCs only.

Cards on Which This Command Runs

PXM45

Syntax

```
dspconsegep <portid>
vpi
[vci]
```

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
vpi	The VPI of the connection.
vci	The VCI of the connection.

Related Commands

cnfoamsegep, dspoamsegep, cnfconsegep, delconsegep

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

dspmbsdft

Display MBS Default

Displays the default MBS configured for the port.

Cards on Which This Command Runs

PXM45

Syntax

dspmbsdft *<portid>*

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.

Related Commands

cnfmbsdft

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the MBS default for port 11:1.1:11.

```
Geneva.7.PXM.a > dspmbsdft 11:1.1:11

           rt-vbr:       nrt-vbr:
  MBS:        1024           1024
```

dspoamsegep

Display OAM Segment Endpoint

Displays whether or not the given port (*portid*) is designated as an OAM Segment Endpoint. See **cnfoamsegep**.

Operation, administration, and maintenance (OAM) is an ATM Forum specification for cells used to monitor virtual circuits. OAM cells provide a virtual circuit-level loopback which demonstrates whether a circuit is up or not.

Cards on Which This Command Runs

PXM45

Syntax

dspoamsegep <*portid*>

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.

Related Commands

cnfoamsegep

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display OAM Segment Endpoint for slot 1, port 1.

```
MGX8850.7.a > dspoamsegep 1.1
```

dsppncon

Display PNNI Connection—display information about an existing call.

Displays information for an active SVC, SPVC, or SPVP. The displayed call information corresponds to the objects described in the **portCallTable** and **portAbrCallTable** MIBs. This command can be issued at any node in the route path.

Cards on Which This Command Runs

PXM45

Syntax

dsppncon <*portid*> <*vpi*> <[*vci*]>

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
vpi	VPI for the call. Default = 0.
vci	VCI for the call. If no VCI is specified, this is a VP connection. Default = 0.

Related Commands

dsppncons

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display VPI/VCI 100 100 on port 1.5.

```
mpgses1.2.PXM.a > dsppncon 1.5 100 100

 Port :            1.5 VPI :    100  VCI :    100
 CallRef:      116  CallRefFlag:   0  CallLeafRef :        0
 Calling-address: 47.009181000000003071f80e4a.000000010500.00
 Calling-subaddress #1: N/A
 Calling-subaddress #2: N/A
 Called-address: 47.009181000000003071f80e49.000000010500.00
 Called-subaddress #1: N/A
 Called-subaddress #2: N/A
 OE Port :            1.7 OE VPI :      1  OE VCI :     39
 OE CallRef:       39  OE CallRefFlag:   0
 OAM-Type : Not an OAM Endpoint
 Connection-type : SPVC   Cast-type : point-to-point   Bearer-class :BCOBX
 Service-category :CBR   Call-clipping-susceptibility:no
 Tx conformance :CBR.1  Rx conformance :CBR.1
 Tx pcr :         50           Rx pcr :         50
 Tx scr :    N/A Rx scr :    N/A
 Tx mbs :    N/A Rx mbs :    N/A
 Tx cdvt : 250000
 Tx frame-discard-option :disable  Rx frame-discard-option :disable
 Max ctd :    N/A
 Max Tx cdv :    N/A    Max Rx cdv :    N/A

Max Tx clr :    N/A    Max Rx clr :    N/A
 NCCI value: 47 00 91 81 00 00 00 00 30 71 f8 0e 4a 00 30 71 f8 0e 4a 01 00 01
```

Non-ABR Call:

```
Port: 5.3   VPI: 33 VCI: 44
CallRef: <call-id>    CallLeafRef: <leaf-id>
Calling-address: <calling AESA address>
Calling-subaddress: <calling AESA sub-address>
Called-address: <called AESA address>
Called-subaddress: <called AESA sub-address>
OE port: <oe_port_id>    OE VPI: <oe_vpi>   OE VCI: <oe_vci>
OE callRef: <oe_callid>
Call-start-time: <timestamp>
Connection-type: SVC    Cast-type: point-to-point    Bearer-class: <bc>
Service-category: CBR   Call-clipping-susceptibility: no
Tx conformance: CBR.1              Rx conformance: CBR.1
Tx pcr: <tx pcr>                   Rx pcr: <rx pcr>
Tx scr: <tx scr>                   Rx scr: <rx scr>
Tx mbs: <tx mbs>                   Rx mbs: <rx mbs>
Tx cdvt: <tx cdvt>                 Rx cdvt: <rx cdvt>
Tx frame-discard-option: disable    Rx frame-discard-option: disable
Max ctd: <max ctd>
Max Tx cdv: <max tx cdv>           Max Rx cdv: <max rx cdv>
Max Tx clr: <max tx clr>           Max Rx clr: <max rx clr>
```

ABR Call:

```
Port: 5.3  VPI: 100  VCI: 200
CallRef: <call-id>    CallLeafRef: <leaf-id>
Calling-address: <calling AESA address>
Calling-subaddress: <calling AESA sub-address>
Called-address: <called AESA address>
Called-subaddress: <called AESA sub-address>
OE port: <oe_port_id>    OE VPI: <oe_vpi>   OE VCI: <oe_vci>
OE callRef: <oe_callid>
Call-start-time: <timestamp>
Connection-type: SVC    Cast-type: point-to-point   Bearer-class: <bc>
Service-category: ABR   Call-clipping-susceptibility: no
Tx conformance: ABR
Tx pcr: <tx pcr>                    Rx pcr: <rx pcr>
Tx mcr: <tx mcr>                    Rx mcr: <rx mcr>
Tx icr: <tx icr>                    Rx icr: <rx icr>
Tx rif: <tx rif>                    Rx rif: <rx rif>
Tx rdf: <tx rdf>                    Rx rdf: <rx rdf>
Tx tbe: <tx tbe>                    Rx tbe: <rx tbe>
Tx nrm: <tx nrm>                    Rx nrm: <rx nrm>
Tx trm: <tx trm>                    Rx trm: <rx trm>
Tx adtf: <tx adtf>                  Rx adtf: <rx adtf>
Tx cdf: <tx cdf>                    Rx cdf: <rx cdf>
Tx frame-discard-option: disable    Rx frame-discard-option: disable
Frtt: <frtt>                        Max ctd: <max ctd>
Max Tx cdv: <max tx cdv>            Max Rx cdv: <max rx cdv>
Max Tx clr: <max tx clr>            Max Rx clr: <max rx clr>
```

dsppncons

Display PNNI Connections

Displays a summary of active calls on one port or all ports. This command can be issued at any node.

Cards on Which This Command Runs

PXM45

Syntax

```
dsppncons
[-port portid]
[-vpi starting-vpi]
[-vci starting-vci]
[-type {p2p | p2mp | ctrl}]
```

Syntax Description

-port	The *portid* represents the PNNI physical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.
-vpi	Keyword that specifies the VPI at which to begin displaying. This keyword can be used only if you specify a *portid*.
-vci	Keyword that specifies the VCI at which to begin displaying. This keyword can only be used if **-vpi** is used.
-type	Keyword that sets the cast type. Only connections of this cast type will be displayed.

Related Commands

dsppncon

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

```
Port       VPI  VCI       CallRef X-Port       VPI  VCI   CallRef  Type OAM-Type
1:1.2:2     1   100       33       1:1.2:2      1   101      32    PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     1   101       32       1:1.2:2      1   100       33   PTP    No
    Calling-Addr:47.00918100000000107be92f3d.000001011802.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.2:2     2   200       34       1:1.6:6      0    49   8388609  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
1:1.6:6     0    49  8388609       1:1.2:2      2   200        34  PTP    No
    Calling-Addr:47.00918100000000107be92f3f.000001011804.00
    Called-Addr:47.00918100000000107be92f3d.000001011802.00
```

dsppnconstats

Display PNNI Connection Statistics

Displays call statistics for a PNNI logical port.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnconstats` *`<portid>`*

Syntax Description

portid	The *portid* represents the PNNI physical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 7-4.

Related Commands

clrpnconstats

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

```
Call Statistics for <portid>
Incoming Call Attempts:     0    Outgoing Call Attempts:     0
Incoming Call Success:      0    Outgoing Call Success:      0
Incoming Call Failures:     0    Outgoing Call Failures:     0
Incoming Filtering Failures:0    Outgoing Filtering Failures:0
Incoming Routing Failures:  0    Outgoing Routing Failures:  0
Incoming CAC Failures:      0    Outgoing CAC Failures:      0
Incoming Timer Failures:    0    Outgoing Timer Failures:    0
Incoming Crankback Failures:0    Outgoing Crankback Failures:0
```

Output Description

Incoming Call Attempts	Number of incoming signaling messages—Setup and AddParty—received on this port for call establishment.
Incoming Call Success	Number of incoming signaling messages—Connect and AddPartAck—received on this port, which indicate successful call establishment.

Incoming Call Failures	Number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port.
Incoming Call Filtering Failures	Number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed the address filtering on this port.
Incoming Routing Failures	Number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there was no route to the destination.
Incoming CAC Failures	Number of incoming point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there were not enough resources as requested in the traffic parameters of the call.
Incoming Timer Failures	Number of signaling timers that timed out on incoming point-to-point and point-to-multipoint SVC/SPVC calls recieved on this port.
Incoming Crankback Failures	Number of crankback IEs recieved on this port for incoming point-to-point and point-to-multipoint SVC/SPVC call attempts.
Outgoing Call Attempts	Number of outgoing signalling messages—Setup and AddParty—sent from this port for call establishment.
Outgoing Call Success	Number of outgoing signaling messages—Connect and AddPartAck—sent from this port, which indicate successful call establishment.
Outgoing Call Failures	Number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port.
Outgoing Call Filtering Failures	Number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed the address filtering on this port.
Outgoing Routing Failures	Number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there was no route to the destination.
Outgoing CAC Failures	Number of outgoing point-to-point and point-to-multipoint SVC/SPVC call attempts that failed on this port because there was not enough resources as requested in the traffic parameters of the call.
Outgoing Timer Failures	Number of signaling timers that timed out on outgoing point-to-point SVC/SPVC calls sent from this port.
Outgoing Crankback Failures	Number of crankback IEs sent from this port for outgoing signaling release messages.

upcon

Up Connection—bring a connection back into service.

Activate a connection that was previously downed by the **dncon** command. The typical use of **dncon** is some form of operational modification or troubleshooting.

Cards on Which This Command Runs

AXSM

Syntax

upcon *<ifNum > <vpi> <vci>*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	Virtual path identifier. On the AXSM, the range is 0–255.
vci	Virtual connection identifier. On t he AXSM, the range is 32–65535 for a VCC. For a VPC, the only *vci* is 0.

Related Commands

dncon

Attributes

Log: log	State: active	Privilege: GROUP1

upport

Up Port

The **upport** command enables (or "ups") a logical port. The usage of **upport** follows **dnport**. See the **dnport** description for information on the use of these commands.

Use **dspports** or **dspport** as needed to determine the port to enable or to see the status of the port after you enable it.

Cards on Which This Command Runs

AXSM

Syntax

upport <*ifNum*>

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.

Related Commands

dspport, **dspports**, **dnport**

Attributes

Log: log	State: active	Privilege: GROUP1

Enable port 1 on the current card.

```
chicago.1.AXSM.a > upport 1
```

Network Optimization Commands

This chapter describes the commands that let you improve the use of network resources. The commands consist primarily of route optimization commands. The chapter begins with a generic description of command syntax.

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is used to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-**num** <*num. conns to del*>] [-**verbose** < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 8-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node use the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to Cisco MGX 8850 Hardware Installation, Rel 2.0.

Table 8-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–2	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2

Connection Capacities of the AXSM

The SVC and SPVC connection capacities for the front card, back card, and physical lines appear in Table 8-2 and Table 8-3. The capacity of a single AXSM card is greater than that of the node itself. Nevertheless, the tables provide these maximums when you plan the use of commands such as **addrscprtn**, **addcon**, and any other command where you may want to know the capacity of the configured item to support connections.

Table 8-2 Maximum Connections by Connection Type and Front Card

Front Card	SVC	SPVC
AXSM-1-2488	128 K	64 K
AXSM-4-622	128 K	64 K
AXSM-16-155	128 K	64 K
AXSM-16-T3E3	128 K	64 K

Table 8-3 Maximum Connections on Back Cards and Lines

Card Type	Back Card Maximum	Physical Line Maximum
OC-48c	128 K	64 K
OC-12c	64 K	32 K
OC-3c	64 K	32 K
T3	64 K	64 K
E3	64 K	64 K

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay
- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 8-4.

Table 8-4 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As the table shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

Slot 1

Bay 2

Line 1

Logical interface 1 (or logical port 1)

cnfpnportloscallrel

Configure PNNI Port Loss of Signal Call Release

The **cnfpnportloscallrel** command lets you shut off the standard delay for rerouting calls on a port when the system detects loss of signal (LOS) on a port.

When the system detects LOS on an NNI link, the switch does not immediately tear down the calls on the link—in case the break is momentary. By default, the system waits for the SSCOP “no-response” and T309 timers to time out before it releases calls on the broken link. The default values for these timers are 30 seconds and 10 seconds, respectively. The system-level assumption (and therefore the default for **cnfpnportloscallrel**) is to retain all the calls for a temporary loss of connectivity, but this can also have the effect of delaying the rerouting of connections. The **cnfpnportloscallrel** command lets you direct the system to reroute calls without delay on a particular port.

Cards on Which Command Executes

PXM45

Syntax

`cnfpnportloscallrel` `<`*`portid`*`> <yes|no>`

Sybtax Description

portid	See PNNI Format, page 8-4. The *portid* represents the PNNI logical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].
yes | *no*	Specifies whether immediate call release is enabled upon LOS. To enable this feature—to remove the standard reroute delay—type “yes.” Default: no.

Related Commands

dsppnportloscallrel

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Enable call release upon LOS for port 3:1.1:1, then confirm its status.

8850_NY.8.PXM.a > **cnfpnportloscallrel** 3:1.1:1 yes

8850_NY.8.PXM.a > **dsppnportloscallrel** 3:1.1:1

Call release on Los :enabled

cnfrrtparm

Configure Reroute Retry Parameters—configures the waiting time intervals for rerouting a failed SPVC.

The **cnfrrtparm** command allows you configure the time periods that the switch waits between each reroute retry attempt.

When an SPVC fails, the system immediately attempts to reroute the connection. If the first reroute attempt fails, the switch keeps trying to reroute the connection according to the *slow retry interval* (**-slowtmr**) and the *fast retry interval base*(**-fasttmrbase**).

The *fast retry interval base* is an incremental value (in 100-millisecond units) that is incremented each time the switch attempts to reroute the connection and fails. The switch then waits the incremented amount of time before it attempts to reroute the connection again. The *fast retry interval base* continues to increment after each reroute attempt until it is equal to the *slow retry interval* value or until the reroute succeeds.

The *slow retry interval* is a fixed value (in seconds) that occurs between all subsequent reroute attempts. After the *fast retry interval base* reaches the *slow retry interval*, the switch attempts to reroute the connection at the rate of the *slow retry interval*. No limit exists for the number of reroute attempts once the *slow retry interval* begins.

For example, if the *fast retry interval base* is 50 100-millisecond intervals (5 seconds) and the *slow retry interval* is 300 seconds (5 minutes), the switch attempts to reroute the connection 5 seconds after the first attempt, 10 seconds after the second attempt, 15 seconds after the third attempt, and so on until the *fast retry interval base* equals 300 seconds (5 minutes). After that, the switch continues to attempt to reroute the connection every 5 minutes or until the reroute is successful.

Cards on Which Command Executes

PXM45

Syntax

cnfrrtparm [-**slowtmr** <*Slow retry interval*>] [-**fasttmrbase** <*Fast retry interval base*]

Syntax Description

-**slowtmr**	The range for *slow retry interval* is 1–300 seconds. The default is 60 seconds. The *slow retry interval* must be greater than *fast retry interval base.*
-**fasttmrbase**	The *fast retry interval base* is a multiplier of 100-millisecond units. The range is 1–3000. The default is 50 100-milliseconds units (5 seconds).

Related Commands

dsprrtparm

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Set the slow timer to 300 second intervals and the fast timer base to 7 seconds (70 x 100 milliseconds). Check the results by executing **dsprrtparm**.

```
8850_NY.7.PXM.a > cnfrrtparm -slowtmr 300 -fasttmrbase 70

8850_NY.7.PXM.a > dsprrtparm

Global SPVC Retry Parameters:
--------------------------------
Slow Retry Interval: 300 sec
Fast Retry Interval Base: 70 (in 100 msec)

8850_NY.7.PXM.a >
```

cnfrteopt

Configure Route Optimization

Configure periodic route optimization to improve bandwidth utilization. This type of optimization is a type of *connection grooming*. To automate route optimization, **cnfrteopt** lets you specify an interval between new optimization cycles—every 2 hours, for example. (To force immediate route optimization, use the **optrte** command.)

You can choose a time period for optimization so that disruption is minimal. For example, you could specify that the switch starts grooming a range of SPVCs for one hour at midnight. Note, however, that route optimization is a background process and does not attempt to optimize all possible connections at once. The load created by route optimization is extremely small and cannot cause congestion.

The nature of SPVCs provides a reason for periodic grooming: during the course of daily operation, better routes may become available. The determining factor for a better route is the maximum cost (*maxcost*). See the **addcon** description for details about the *maxcost* parameter. The PNNI protocol identifies this maximum cost by another name: *administrative weight* (AW).

Note If you do not specify a *maxcost* with either the **addcon** or **cnfcon** command, the routing protocol uses the AW on only forward links to calculate a new route for the connection. If the connection has a specified maxcost, the routing protocol calculates possible routes by using the AW in both directions.

Usage Guidelines

Note the following characteristics of route optimization:

- Within a range of connections, the **cnfrteopt** command applies to only the master endpoints. The slave endpoints are not processed by **cnfrteopt**.
- Route optimization applies to only routed connections. The switching fabric does load comparison between the routing cost of a connection's current route and the new, potential, best route.

By default, the PXM45 calculates that a route is better if its routing cost is 30% less than the current cost. You can change this cost threshold through the **cnfrteoptthld** command.

The following briefly characterizes the defaults for **cnfrteopt**:

- The default state is disabled.
- If you do not specify a range, all connections on the port are subject to optimization.
- If you not specify an interval, optimization begins every 60 minutes.
- If you do not specify a time of day, the default is any time during the day (but still subject to the interval of minutes between optimization commencement).

Cards on Which Command Executes

PXM45

Syntax

```
cnfrteopt
<portid>
[{enable | disable}]
```

```
[-range <starting-vpi/vci..ending-vpi/vci>]
[-interval <interval>]
[-tod <start-time..end-time>]
```

Syntax Description

portid	Port identifier has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].
enable | disable	Enables or disables route optimization. The default is disabled, but if grooming is operational and you want to disable it, you must execute **cnfrteopt** and enter "disable."
-range	Keyword that specifies a range of connections for grooming. Use the notation as it appears on the syntax line: type a slash between the VPI and VCI and two dots with no spaces between the starting and ending values. For example, 100/1000..200/10000 is a valid parameter to follow the –**range** keyword. The ranges are: • The *vpi* range is 0–4095. • The *vci* range is 32–65535. The VPI of the starting SPVC must be less than the ending VPI, and the starting VCI must be less than the ending VCI. Note that the default range is *all* connections on the port specified by *portid*. Therefore, if you want to groom all connections on the *portid*, simply leave out the -**range** command delineator.
-interval	Keyword that specifies the frequency at which grooming begins. The units of measure are minutes. The range is 10–10000. The default is 60. Counting starts at one of two moments: • The moment you execute **cnfrteopt** • The starting time specified by TOD in **cnfrteopt** If the interval is less than half the amount of time specified by the *start-time..end-time* parameter, route optimization may begin more than once during the time period. For example, if the periods of optimization are two hours beginning at midnight and 4:00 AM and the interval is one hour, route optimization could occur two to four times per day.
-tod	Keyword that specifies the time to start and stop grooming. The format is a 24-hour clock: 00:00–23:59. The default for both *start* and *end-time* is 00:00. If you execute **cnfrteopt** during the time specified by **tod**, the optimization cycle begins during the next time interval. If the time for the node changes (by way of the **cnftime** command, for example), the node might skip one optimization cycle. **Note** Use the notation in the Syntax section: type two dots with no spaces between starting and ending times.

Cards on Which Command Executes

PXM45

Related Commands

cnfrteoptthld, **optrte**, **dsprteoptcnf**, **dsprteoptstat**

Attributes

Log: log	State: active	Privilege: GROUP1

Examples

For logical port 2 on the lower bay of the service module in slot 1, configure 1 hour of connection grooming starting between 1:00 and 3:00 AM local time. The range of SPVCs is 100.1000 through 100.10000.

```
cnfrteopt 1:2.1:2 enable -range 100/1000..100/10000 -interval 60 -tod 01:00..03:00
```

cnfrteoptthld

Configure Route Optimization Threshold

Configure a threshold the system uses to decide whether one route is a sufficient improvement to warrant re-routing. The criterion for selecting a new route is a threshold in the form of the percent of difference in route cost.

The default for route optimization is a 30% reduction of the cost of a route. Using this mechanism, the PXM45 selects a given route if it costs 30% less than the cost of the current route. With **cnfrteoptthld**, you can change the percentage of routing cost-improvement.

Syntax

```
cnfrteoptthld <percent>
```

Syntax Description

percent	The percent of reduction in routing cost that triggers re-routing. The range is 5–100. The default is 30.

Cards on Which Command Executes

PXM45

Related Commands

cnfrteopt, **optrte**, **dsprteoptcnf**, **dsprteoptstat**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Change the re-routing threshold to a 20% reduction in the route cost.

```
pinnacle.7.PXM> cnfrteoptthld 20
```

dspload

Display Load—displays maximum channels and bandwidth and currently available channels and bandwidth on a partition.

This command displays the configured maximum and guaratneed number of channels and the configured maximum and guaranteed bandwidth on a partition on a logical interface. It also displays the currently available number of channels and the currently available bandwidth on the partition.

This information can allow you to determine the throughput requirements for a partition and the actual resource utilization.

The output shows the configured bandwidth for user connections of all service classes.

The display also shows the load created by control VCs. The control VCs are of three types: SSCOP, PNNI-RCC, and ILMI (if ILMI is enabled). The bandwidth allocations for SSCOP and PNNI-RCC control VCs are configurable (see **cnfpnctlvc**.) The bandwidth for an ILMI VC is fixed, as follows: PCR=1000 cps; SCR=50cps; and MBS=1024 cells.

Cards on Which This Command Runs

AXSM

Syntax

`dspload` `<`*`ifNum`*`> <`*`partId`*`>`

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
partId	The partition identifier. The range is 1–20. If necessary, use **dsprscprtns** to see the existing partitions.

Related Commands

dsprscprtn, **addcon**, **dspcons**, **dspcon**, **cnfcon**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the load on partition number 1 on logical port 1.

```
MGX8850.1.AXSM.a > dspload 1 1
+-------------------------------------------+
|   I N T E R F A C E   L O A D   I N F O   |
+-------------------------------------------+
| Maximum Channels         : 0001000         |
| Guaranteed Channels      : 0000000         |
| Igr Maximum Bandwidth    : 0353208         |
| Igr Guaranteed Bandwidth : 0176604         |
| Egr Maximum Bandwidth    : 0353208         |
| Egr Guaranteed Bandwidth : 0176604         |
| Available Igr Channels   : 0000998         |
| Available Egr Channels   : 0000998         |
| Available Igr Bandwidth  : 0351708         |
| Available Egr Bandwidth  : 0351708         |
+-------------------------------------------+
|          E X C E P T -- V A L U E S        |
+-------------------------------------------+
| SERV-CATEG | VAR-TYPE | INGRESS  | EGRESS  |
| VSI-SIG    | Avl Chnl | 0000998  | 0000998 |
| CBR        | Avl Chnl | 0000998  | 0000998 |
| VBR-RT     | Avl Chnl | 0000998  | 0000998 |
| VBR-nRT    | Avl Chnl | 0000998  | 0000998 |
| UBR        | Avl Chnl | 0000998  | 0000998 |
| ABR        | Avl Chnl | 0000998  | 0000998 |
+-------------------------------------------+
| VSI-SIG    | Avl Bw   | 0351708  | 0351708 |
| CBR        | Avl Bw   | 0351708  | 0351708 |
| VBR-RT     | Avl Bw   | 0351708  | 0351708 |
| VBR-nRT    | Avl Bw   | 0351708  | 0351708 |
| UBR        | Avl Bw   | 0351708  | 0351708 |
| ABR        | Avl Bw   | 0351708  | 0351708 |
+-------------------------------------------+
```

Table 8-5 Interface Load Info Descriptions

Field	Descriptions
Maximum Channels	The total maximum number of channels available on the interface.
Guaranteed Channels	The maximum number of channels that are guaranteed to be available at the maximum bandwidth.
Igr Maximum Bandwidth	The maximum bandwidth available on the interface in the ingress direction.
Igr Guaranteed Bandwidth	The maximum bandwidth that is guaranteed to be available in the ingress direction when the maximum number of guaranteed channels are in use.
Egr Maximum Bandwidth	The maximum bandwidth available on the interface in the egress direction.
Egr Guaranteed Bandwidth	The maximum bandwidth that is guaranteed to be available in the egress direction when the maximum number of guaranteed channels are in use.
Available Igr Channels	The number of ingress channels that are currently available.
Available Egr Channels	The number of egress channels that are currently available.

Table 8-5 Interface Load Info Descriptions (continued)

Field	Descriptions
Available Igr Bandwidth	The amount of bandwidth that is currently available in the ingress direction.
Available Egr Bandwidth	The amount of bandwidth that is currently available in the egress direction.
EXCEPT -- VALUES	The Except Values display the currently available bandwidth (Avl Bw) and currently available channels (Avl Chnl) for each class of service: VSI-SIG, CBR, VBR-RT, VBR-nRT, UBR, ABR.
SERV-CATEG	Service Category: The Class Of Service: VSI-SIG, CBR, VBR-RT, VBR-nRT, UBR, ABR.
VAR- TYPE	Variable Type: The available bandwidth (Avl Bw) or the available channels (Avl Chnl).
VSI - SIG	Virtual Switch Interface Signaling:
CBR	Constant Bit Rate: Used for connections that require a high QoS and strict cell delay variation (CDV).
VBR-RT	Variable Bit Rate–Real Time: Used for connections that have burst traffic and that require a strict cell delay variation (CDV).
VBR-nRT	Variable Bit Rate–non-Real Time: Used for connections that do not require end to end timing.
UBR	Unspecified Bit Rate: Used for connections that can allow any amount of data, up to a specified maximum, to be transmitted, but with no guarantees in terms of cell loss rate and delay.
ABR	Available Bit Rate: Used for connections that do not require timing relationships between source and destination endpoints. ABR provides no guarantees in terms of cell loss or delay, and provides only a best-effort service. Cell rates are adjusted in response to the state or condition of the network and its ability to successfully deliver data.

dsppnportloscallrel

Display PNNI Port Loss of Signal Call Release

This command displays the enable status of the LOS call release feature. See **cnfpnportloscallrel** for a description of this feature.

Cards on Which Command Executes

PXM45

Syntax

`dsppnportloscallrel` <*portid*>

Sybtax Description

portid	See PNNI Format, page 8-4. The *portid* represents the PNNI logical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*].

Related Commands

cnfpnportloscallrel

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

First, confirm that LOS call release is disabled on port 3:1.1:1. Enable it, then confirm that it’s enabled.

8850_NY.8.PXM.a > **dsppnportloscallrel** 3:1.1:1

Call release on Los :disabled

8850_NY.8.PXM.a > **cnfpnportloscallrel** 3:1.1:1 yes

8850_NY.8.PXM.a > **dsppnportloscallrel** 3:1.1:1

```
Call release on Los :enabled
```

dsprrtparm

Display Reroute Parameters—displays the current values for two types of reroute intervals.

The **dsprrtparm** command shows the current slow interval time and fast timer base. See the description of **cnfrrtparm** for the function of these intervals.

Cards on Which Command Executes

PXM45

Syntax

dsprrtparm

Syntax Description

This command takes no parameters.

Related Commands

cnfrrtparm

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current, global reroute parameters for SPVCs. The parameters are the defaults.

```
8850_NY.7.PXM.a > dsprrtparm

Global SPVC Retry Parameters:
--------------------------------
Slow Retry Interval: 60 sec
Fast Retry Interval Base: 50 (in 100 msec)
```

dsprteoptcnf

Display Route Optimization Configuration

Display the current configuration for route optimization. The configuration itself originates with the **cnfrteopt** command. The **dsprteoptcnf** display shows the following:

- The node-level threshold for route optimization. It is the percent of reduction in the route cost.
- Identity of the optimization target by port and VPI/VCI range.
- Enable status of optimization.
- Interval between times that optimization begins.
- Start and stop times for route optimization.

Cards on Which Command Executes

PXM45

Syntax

`dsprteoptcnf`

Syntax Description

This command takes no parameters.

Related Commands

cnfrteopt, **opte**, **dsprteoptstat**, **cnfrteoptthld**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current route optimization for the switch. No optimization has been configured on any ports.

```
pop20two.7.PXM.a > dsprteoptcnf
Configuration of Route Optimization:
Percentage Reduction Threshold: 30
Port          Enable   VPI/VCI Range    Interval   Time Range
1:2.1:2       no
1:2.2:3       no
```

dsprteoptstat

Display Route Optimization Status

Display the current percent of route cost reduction. This percent is a threshold that the PXM45 requires to determine that one route costs sufficiently less to warrant re-routing. The percent applies to all connections on the node. The system default is 30%, but you can configure a percent through the **cnfrteoptthld** command. For more details on route optimization, see the **cnfrteopt** description.

Cards on Which Command Executes

PXM45

Syntax

dsprteoptstat

Syntax Description

This command takes no parameters.

Related Commands

cnfrteopt, **cnfrteoptthld**, **optrte**, **dsprteoptstat**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the current route optimization status for the switch. The display shows the default optimization of 30%.

```
pop20two.7.PXM.a > dsprteoptstat
Configuration of Route Optimization:
Percentage Reduction Threshold: 30
```

optrte

Optimize Routes

Force immediate optimization of either a single SPVC route, a range of SPVCs, or all SPVCs on a particular port. (Connection *grooming* is a common word for optimization.)

Re-routing depends on a reduction in the cost of the route. If the PXM45 can find a route with sufficiently lower cost, the SPVC is de-routed then re-routed. The system default is a 30% reduction in the cost but is configurable through the **cnfrteoptthld** command. For a detailed explanation of route optimization, see the description of **cnfrteopt**.

Cards on Which Command Executes

PXM45

Syntax

```
optrte <portid>
[-vpi <vpi>]
[-vci <vci>]
[-range <starting-vpi/vci..ending-vpi/vci>]
```

Syntax Description

portid	Port identifier has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*].
-vpi	Keyword that specifies the *vpi*. The range is 0–4095.
-vci	Keyword that specifies the *vci.* The range is 32–65535.
-range	Keyword that specifies the range of connections for grooming. The VPI of the starting SPVC must be less than the ending VPI, and the starting VCI must be less than the ending VCI. Use the notation as it appears on the syntax line: type a slash between the VPI and VCI and two dots with no spaces between the starting and ending values. For example, 100/1000..200/10000 is a valid parameter to follow the **–range** keyword. The ranges are: • The *vpi* range is 0–4095. • The *vci* range is 32–65535. Note that the default range is *all* connections—on the entity specified by *portid*. Therefore, if you want to groom all connections on the *portid*, simply leave out the -**range** command delineator.

Related Commands

cnfrteopt, **cnfrteoptthld**, **dsprteoptcnf**, **dsprteoptstat**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Immediately find a better route for the SPVC with vpi.vci 1000.50000 on portid 1:2.1:1

```
pop20two.7.PXM.a > optrte -vpi 1000 -vci 50000
```

routeShow

Route Show

Show the current IP routing of the network layer of the operating system.

Cards on Which This Command Runs

PXM45

Syntax

routeShow

Related Commands

routestatShow

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the current IP routing of the network layer of the operating system.

```
pinnacle.8.PXM.a > routeShow

ROUTE NET TABLE
destination      gateway          flags  Refcnt  Use           Interface
------------------------------------------------------------------------
0.0.0.0          172.29.23.149    1      1       21778         lnPci0
0.0.0.0          172.29.23.1      3      0       2755          lnPci0
172.1.1.0        172.1.1.149      1      0       0             atm0
172.29.23.0      172.29.23.149    1      2       5275          lnPci0
------------------------------------------------------------------------

ROUTE HOST TABLE
destination      gateway          flags  Refcnt  Use           Interface
------------------------------------------------------------------------
0.0.0.0          0.0.0.0          5      0       0             sl0
127.0.0.1        127.0.0.1        5      1       0             lo0
172.29.23.3      172.1.1.149      5      0       3555          atm0
172.29.23.5      172.1.1.149      5      0       3304          atm0
172.29.23.7      172.1.1.149      5      0       3335          atm0
171.71.29.18     172.1.1.149      5      0       3304          atm0
172.29.23.18     172.1.1.149      5      0       3304          atm0
172.29.23.28     172.1.1.149      5      0       6127          atm0
172.29.23.29     172.1.1.149      5      1       6065          atm0
171.71.29.32     172.1.1.149      5      0       5842          atm0
171.71.29.44     172.1.1.149      5      0       3304          atm0
172.29.23.53     172.1.1.149      5      0       3304          atm0
171.71.29.59     172.1.1.149      5      0       3304          atm0
171.71.28.126    172.1.1.149      5      0       3309          atm0
------------------------------------------------------------------------

pinnacle.8.PXM.a >
```

routestatShow

Show Routing Statistics

Display statistics for the current IP routing in the network layer of the system.

Cards on Which This Command Runs

PXM45

Syntax

routestatShow

Related Commands

routeShow

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Display the current IP routing statistics.

```
pinnacle.8.PXM.a > routestatShow

routing:
        0 bad routing redirect
        0 dynamically created route
        0 new gateway due to redirects
        0 destination found unreachable
        11095 uses of a wildcard route

pinnacle.8.PXM.a >
```

rrtcon

Re-route Connection—force the system to re-route a connection

The **rrtcon** command lets you trigger the immediate re-routing of a connection.

Cards on Which This Command Runs

AXSM

Syntax

rrtcon *<ifNum> <vpi> <vci>*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	The VPI of the connection. For UNI, the range is 0–255. For NNI, the range is 0–4095.
vci	The VCI of the connection. For a VCC, the *vci* range is 32–65535. For a VPC, the VCI is always 0.

Related Commands

dspcons, dspcon

Attributes

Log: log	State: active	Privilege: GROUP1

Troubleshooting Commands

This chapter describes commands that directly or indirectly facilitate troubleshooting. They include commands for viewing and clearing alarms and statistics. The chapter begins with a description of the CLI, various elements of the AXSM, and the PNNI port identifier.

Position-Dependent and Keyword-Driven Parameters

A command can include parameters that are *keyword-driven* or *position-dependent*.

For position-dependent parameters, you must type parameters in the order they appear in the syntax description or on-line help. To create a logical port, for example, the position-dependent syntax is:

addport <*ifNum*> <*bay.line*> <*guaranteedRate*> <*maxrate*> <*sctID*> <*ifType*> [*vpi*]

For a keyword-driven parameter, a keyword must precede the value. The keyword is preceded by a dash and followed by the parameter (–**timeout** <*secs*>, for example). The order you enter keyword-driven parameters does not matter—although any preceding or succeeding, position-dependent parameters must appear as they do in the command syntax description.

In the following syntax example, the command is to delete more than one connection at a time. The mandatory, position-dependent connection identifier consist of a logical port (*ifNum*) and the VPI and VCI of the first connection to delete. After the connection identifier, the line shows two optional, keyword-driven parameters. These keyword-driven parameters let you enter the number of connections to delete and specify verbose mode:

delcons <*ifNum*> <*vpi*> <*vci*> [-num <*num.conns to del*>] [-verbose < 1 | 0 >]

Command Entry

When you enter a command with the current version of the product, you must type all intended arguments before you press the Return key or Enter key.

If you press the Return key or Enter key with incorrect parameters or no parameters (if the command requires parameters), a message displays the syntax and parameter ranges. The returned message may also suggest what the problem is. For example, the message may warn of too few parameters. No error messages or warnings appear until you complete the command.

Identifying the AXSM Models

The model number of an AXSM identifies the line speed, line count, and number of bays (see Table 9-1.) Note that the number of lines applies to an individual back card, so the total number of lines supported by the front card equals the highest line number times the number of bays. The OC-48 card AXSM-1-2488 has the lowest number of lines—one. The highest number of lines exist on the AXSM-16-155 and AXSM-16-T3E3—16, as the name indicates.

The MGX 8850 node use the concept of a *bay*. The bay refers to the upper or lower location of a single-height card. (The switch has a double-height card cage, so a single-height back card necessarily occupies either an upper or lower position.)

The T3/E3, OC-3, and OC-12 versions of the AXSM can have two back cards, one in bay 1 (upper location of the back slot) and the second in bay 2 (lower slot). The MGX-AXSM-1-2488 (OC-48 AXSM) can have a back card in bay 1 only. For further descriptions and illustrations of the card sets, refer to *Cisco MGX 8850 Hardware Installation, Release 2.1*.

Table 9-1 Valid Line Numbers and Number of Bays for AXSM Card Types

Front Card	Speed	Lines	Bays
AXSM-1-2488	OC-48	1	1
AXSM-4-622	OC-12	1–4	1–2
AXSM-16-155	OC-3	1–8	1–2
AXSM-16-T3E3	T3, E3	1–8	1–2
AXSM-2-622-E	OC12	1	1–2
AXSM-8-155-E	OC3	1–4	1–2
AXSM-16-T3E3-E	T3, E3	1–8	1–2

Identifying Physical and Logical Elements

The Private Network-to-Network Interface (PNNI) control protocol and the service modules use different formats to identify the same entity. For example, the format of a logical port that you enter on an AXSM is different from the format you would enter on the PXM45. This section describes these formats in the PNNI and AXSM contexts and how they correspond to each other. The parallel actions of configuring or displaying logical elements on different cards is broadly illustrated in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

Apart from the way PNNI and the lower levels of logic identify the same element, the sequence of commands also needs explanation. When you configure logical ports—for just one example—you must complete certain tasks on the AXSM CLI before or after related PNNI tasks. For certain commands, this manual lists prerequisite commands or tasks. For more details on the sequence of tasks, refer to the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0, for more details of this sequence.

AXSM Format

On a service module, you identify the follow when you provision the capabilities of the card:

- Slot
- Bay

- Line
- Logical port
- Port group
- Resource partition

Not all of these elements correspond to elements you specify on the PXM45. Subsequent paragraphs describe only the common elements that are visible on the CLI of the PXM and the service module. The preceding elements are further defined in the *Cisco MGX 8850 Routing Switch Software Configuration Guide*, Release 2.0.

For a UNI or NNI, one logical interface (or logical port) exists per physical line. For virtual network to network interfaces (VNNIs), you can configure multiple ports on a line. The maximum number of logical ports on an AXSM is 60 regardless of the AXSM model or the number of lines on the back cards. The range of logical port numbers is 1–60 for an AXSM regardless of whether the interface type is UNI, NNI, or VNNI.

PNNI Format

The PNNI controller requires the following format to identify a physical port:

[*shelf.*]*slot*[:*subslot*].*port*[:*subport*]

The PNNI physical *port identifier* (physical port ID) consists of a series of mandatory elements. Note the period or colon associated with each element inside the square brackets. The elements of the physical port ID are as follows:

- The *shelf* is always 1 for the current product and so is usually omitted.
- The *slot* number of the front card.
- *Subslot* is the number of the bay where the back card resides. This number is 1 or 2.
- *Port* is the physical line.
- *Subport* corresponds to the resource partition on the AXSM. For a UNI or NNI, this resource partition is the same number as the logical port number (*ifNum*) on the AXSM. For a virtual network-to-network interface (VNNI), the number does not directly correspond to the

For each physical port number, PNNI also generates a logical port number as an encrypted form of the physical port number. The logical port number appears as an unformatted numerical string. For example, a PNNI physical port ID may have the form 1:1.2:2, so the PNNI logical port number would be 16848898. Where needed, the descriptions in the PNNI command chapter define the need for this logical port number. (This section does not define a PNNI logical port number, nor does it describe the correspondence between an AXSM port and a PNNI logical port number.) For the correspondence between a PNNI physical port and the port identifier on an AXSM, see Table 9-2.

Table 9-2 Mapping PNNI Port ID to AXSM Elements

PNNI port	AXSM
Shelf	N/A
Slot	Slot
Subslot	Bay (for upper or lower back card)
Port	Line
Subport	Logical interface (*or port*)

As Table 9-2 shows, a port to PNNI is a line on the AXSM, and a subport to PNNI is a logical interface (or logical port) on an AXSM. An example of a PNNI physical port identifier is 1:2.1:1. This *portid* corresponds to an AXSM, with the following particulars:

- Slot 1
- Bay 2
- Line 1
- Logical interface 1 (or logical port 1)

abortofflinediag

Abort Offline Diagnostics

Aborts the currently running offline diagnostics.

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

```
abortofflinediag
<slot>
```

Syntax Description

slot	The slot of the card for which to abort the offline diagnostics.

Cards on Which This Command Runs

PXM45

Related Commands

cnfdiag, cnfdiagall, dspdiagcnf

Attributes

Log: no	State: active, standby	Privilege: SERVICE_GP

Caution If offline diagnostic is running on the AXSM, the AXSM must be in the standby state.

Example

```
abortofflinediag 1
```

addchanloop

Add Channel Loopback—add a loopback to a specific channel in a network test configuration.

The channel loopback tests the integrity of the connection (channel) at the local UNI or across the network. The system returns an error message if the connection is broken or incorrect data arrives at the end of the loopback. The maximum number of connection loopbacks that can exist on an AXSM is 256.

The **addchanloop** command applies to a network that is not carrying live traffic because the test is totally intrusive. The test requires some form of a testing device to generate a cell stream. The parameters for such a stream are the number of cells transmitted through the loop, the cell transfer rate, and so on. (To test connection integrity in a non-destructive way while the connection carries user data, use **tstdelay** on the ingress or **tstconseg** on the egress. These commands generate one OAM cell for each command execution.)

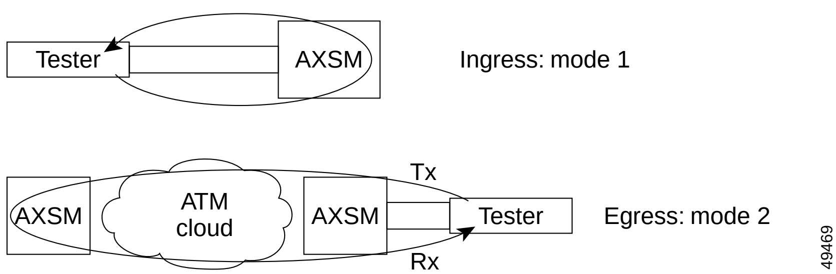
A connection can have only one loopback at a time. Therefore, you cannot add a loopback for both directions at the same time. The loopback remains until you delete it by executing **delchanloop**. To see the presence of connection loopbacks on a per-port basis, use **dspchanloop**.

The **addchanloop** command lets you specify the direction of cell flow within the loop (see Figure 9-1):

- In the ingress direction, the cells travel from the tester to the queueing engine on the AXSM; then back to the tester.
- In the egress direction, the cells travel from the tester to the local AXSM; then across the network to the remote AXSM. At the far end, the cells go to the queueing engine then return back across the network to the tester.

The maximum number of loopbacks that can exist on an AXSM is 256.

Figure 9-1 Connection (Channel) Loopbacks on the Ingress and Egress



Cards on Which This Command Runs

AXSM

Syntax

`addchanloop` *`<ifNumber> <vpi> <vci> <loopback mode>`*

Syntax Description

ifNumber	The logical port number. The range for AXSM is 1–60.
vpi	The VPI of the connection. The range is 0–4095.

vci	The VCI of the connection. The range is 32–65535.
loopback mode	The mode of the loopback is the direction. • 1 = the ingress direction. • 2 = the egress direction.

Related Commands

delchanloop, **dspchanloop**

Tabulates

Log: log	State: active, standby	Privilege: SERVICE_GP

Example

Add a loopback on the connection with VPI/VCI of 1 50 on logical port 4. No message is returned unless an error occurs in command execution (such as an attempt to add a channel loopback to a connection that already has a loopback).

```
pop20two.1.AXSM.a > addchanloop 4 1 50
```

Check for the presence of the loopback by displaying all channel loopbacks on port 4.

```
pop20two.1.AXSM.a > dspchanloop 4
Port     Type      lVPI      lVCI      rVPI      rVCI
4     igrLpbk       1        50         0        35
```

addlnloop

Add Line Loop

Specifies a loopback state for a line on the current service module.

Note Before you can change the loopback type for an existing loopback, you must first delete the loopback by executing **dellnloop** or just **addlnloop** with the No loopback mode.

Cards on Which This Command Runs

AXSM

AXSM Syntax

```
addlnloop <-ds3 | -sonet bay.line> <-lpb loopback type>
```

Note For AXSM cards, the keyword **ds3** applies to both T3 and E3 line types.

AXSM Syntax Description

-ds3 **-sonet**	Specifies a DS3 line (E3 or T3) or a SONET line (OC-3c, OC-12c, OC-48c).
bay.line	Identifies the bay (1 or 2) and the number of the line. The line number can be 1 to the highest numbered line on the back card.
-lpb	Specifies the loopback type for the line type. The entry for no loopback (1) removes any existing loopback. 1 = No loopback 2 = Local loopback 3 = Remote loopback

Related Commands

dellnloop

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Adding a DS3 line in a loopback state.

```
MGX8850.1.11.AXSM.a > addlnloop -ds3 1.1 -lpb 2
```

clrallcnf

Clear All Configurations

Clear all configurations for all the cards in the node. After you enter the command, the system prompts you to confirm the action.

Caution Be absolutely sure you need to execute this command because it clears all configuration files on the PXM45. After **clrallcnf**, you need to reconfigure the switch.

Cards on Which This Command Runs

PXM45

Syntax

`clrallcnf`

Related Commands

restoreallcnf

Attributes

Log: log	State: active, init	Privilege: SERVICE_GP

Example

Clear all the configuration elements for all the cards in the node.

```
node1.7.PXM.a > clrallcnf
All SM's config will be deleted, and
        the shelf will be reset.
Do you want to proceed (Yes/No)?
```

clralmcnt

Clear Alarm Counters

Clear all the alarm counters and statistics on the specified line on the current card. All counters are reset to 0. All statistical alarms that are displayed by **dspalms** and **dspalmcnt** are cleared. The system does not display a response unless it detects a syntax error.

Cards on Which This Command Runs

AXSM

Syntax

clralmcnt <*bay.line*>

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The line number can be 1 to the highest numbered line on the back card.

Related Commands

dspalmcnt

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear the alarms on line 1 or the lower back card.

```
node1.1.2.AXSM.a > clralmcnt 2.1
```

clrbecnt

Clear Bit Error Count

The **clrbecnt** command lets you clear the APS-related bit error counters for a working line. To see the contents of the error counters, use the **dspbecnt** command.

Cards on Which This Command Runs

AXSM

Syntax

`clrbecnt` `<`*`working-bay.line`*`>`

Syntax Description

working-bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 9-1.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, dspbecnt

Attributes

Log: no log	State: active	Privilege: SUPER_GP

clrcdcnt

Clear Card Counters

Clears the counters for received and transmitted cells on the current card. See **dspcdcnt** for examples of the counter contents. The information that **clrcdcnt** clears and that **dspcdcnt** displays primarily applies to debugging.

Cards on Which This Command Runs

AXSM

Syntax

```
clrcdcnt
```

Syntax Description

This command takes no parameters.

Related Commands

dspcdcnt, dspchancnt

Attributes

Log: log	State: active	Privilege: SUPER_GP

clrchancnt

Clear Channel Counters—clears the error counters for a single connection.

Clear all counters for ATM cells on a connection (channel). The command applies to an SVC or an SPVC. For a list of displayed counters, see the example of **dspchancnt**. Once you execute **clrchancnt**, the previous counter contents are unrecoverable.

Note This command does not apply to OC-48 cards.

Cards on Which This Command Runs

AXSM

Syntax

`clrchancnt` *`<ifNum> <vpi> <vci>`*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	The VPI has the range 0–255 for a UNI or 0–4095 for a UNI or VNNI.
vci	The VCI in the range 1–65535.

Related Commands

dspchancnt

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear all the connection counters on AXSM for connection 100.1000 on logical port 3.

```
node1.1.AXSM.a > clrchancnt 3 100 1000
```

clrchancnts

Clear Channel Counters—clears the error counters for all connections on a card.

Clears the statistics counters on all connections.

Cards on Which This Command Runs

AXSM

Syntax

`clrchancnts`

Syntax Description

No parameters.

Related Commands

dspchancnt, clrchancnt

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

```
SunnyVale.13.AXSM.a > clrchancnts
```

clrdiagerr

Clear Diagnostics Errors

Clears all diagnostics error messages that are currently in memory.

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

```
clrdiagerr
<slot>
```

Syntax Description

slot	The slot of the card for which to clear the diagnostics errors.

Cards on Which This Command Runs

PXM45

Related Commands

dspdiagerr

Attributes

Log: No	State: active, standby	Privilege: SERVICE_GP

Example

```
clrdiagerr 7
```

clrdiagstat

Clear Diagnostics Statistics

Clears all the diagnostics statistics currently in memory. The diagnostics statistics program keeps count of how many times diagnostics has run.

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

clrdiagstat <*slot*>

Syntax Description

slot	The slot of the card for which to clear the diagnostics statistics.

Cards on Which This Command Runs

PXM45

Related Commands

dspdiagstat

Attributes

Log: no	State: active or standby	Privilege: SERVICE_GP

Example

```
clrdiagstat 7
```

clrerr

Clear Error

Clear all error log files for a slot. After you execute **clrerr**, the information is unrecoverable. The cleared information consists of system-level or internal errors and so applies more to developers and individuals capable of internal troubleshooting. To see the information that **clrerr** removes, see **dsperr**.

After you enter **clrerr**, the system prompts you to confirm that you want to clear all error log files.

Cards on Which This Command Runs

PXM45

Syntax

```
clrerr <-s1 slot>
```

Syntax Description

-sl	Number of the slot. The value of *slot* is any slot in the switch.

Related Commands

dsperr

Attributes

Log: no log	State: active, standby	Privilege: SUPER_GP

Example

Clear all error log files on the PXM45.

```
pinnacle.7.PXM.a > clrerr
Do you want to clear error log file (Yes/No)?
```

clrerrhist

Clear Error History—clear the history of errors for a card.

The **clrerrhist** commands resets the contents of the error history file for a particular card. Although you execute this command on the CLI of the PXM45, you can specify the error history of any slot. For a list of the information fields in an error history file, see the description of **dsperrhist**.

Cards on Which This Command Runs

PXM45

Syntax

```
clrerrhist [slot]
```

Syntax Description

slot	Number of the slot—any slot in the switch. If you do not enter a slot number, the system clears the error history on the current PXM45.

Related Commands

dsperrhist

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Clear the error history for the current PXM45. The system returns a message indicating whether the operation succeeded.

```
pop20one.7.PXM.a > clrerrhist 7
Log of Errors and Failures for slot# 7 is cleared.
```

clrilmicnt

Clear ILMI Counters

Clears the ILMI statistics for a partition and logical port on a service module.

Cards on Which This Command Runs

AXSM

Syntax

clrilmicnt *<ifNum>* *<partId>*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
partId	Number of the partition in the range 1–20.

Related Commands

dspilmicnt, dspilmi, dspilmis

Attributes

Log: no log	State: active	Privilege: SUPER_GP

Example

Clear the ILMI statistics for logical interface 1, resource partition 1. Before doing so, confirm the existence of these entities by executing **dspparts**.

```
pop20two.1.AXSM.a > dspparts
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
-----------------------------------------------------------------------------
 1   1    2   10000   10000   10000   10000   10  100   100  1000      0     10

pop20two.1.AXSM.a > clrilmicnt 1 1
ilmi stats for ifNum 1, partId 1 cleared
```

clrlncnt

Clear Line Counters—clear cell and connection-related counters for a line.

See **dsplncnt** for descriptions of the counters. The system returns a response only if an error occurs.

Cards on Which This Command Runs

AXSM

Syntax

`clrlncnt` *`<bay.line>`*

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The range for *line* can be 1 to the highest numbered line on the back card.

Related Commands

dsplncnt

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear the line counters for line 1 in bay 1 on the current AXSM.

```
wilco.1.AXSM.a > clrlncnt 1.1
```

clrlog

Clear Log

Use the **clrlog** command to clear either a specific log file or all log files. The log resumes accumulating event messages after the command executes.

Cards on Which This Command Runs

PXM45

Syntax

clrlog [**-log** <*log*>]

Syntax Description

-log	Specifies the type of log file (*log*) to clear. See **dsplog** for a list of the types of logs files.

Related Commands

dsplog, **dsplogs**

Attributes

Log: log	State: active, standby	Privilege: SUPER_GP

Example

Clear all event log files on the PXM45 card.

```
wilco.7.PXM.a > clrlog
```

clrportcnt

Clear Port Counter

Clear counter values on a specific logical port.

Cards on Which This Command Runs

AXSM

Syntax

clrportcnt <*ifNum*>

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.

Related Commands

clrportcnts, **dspportcnt**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear all the port counters on port 1.

```
flyers01.17.AXSM.a > clrportcnt 1
```

clrportcnts

Clear Port Counters

Clear all port counters on the current AXSM. The system does not return a message unless a syntax error occurs (such as a spurious character following the command on the CLI).

Cards on Which This Command Runs

AXSM

Syntax

clrportcnts

Syntax Description

This command takes no parameters.

Related Commands

clrportcnt, **dspportcnt**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear all the port counters on the current AXSM.

```
flyers01.17.AXSM.a > clrportcnts
```

clrxbaralm

Clear Crossbar Alarm—clear the crossbar alarms.

To see the alarms that **clrxbaralm** clears, execute **dspxbaralarm** or **dspswalms**. The **clrxbaralm** command clears the alarms for either a specific switch plane or for all the switch planes on the active switching card. In an MGX 8850 node, the switching slot is the slot number of the PXM45.

Cards on Which This Command Runs

PXM45

Syntax

clrxbaralm < * | [*slot plane*]>

Syntax Description

*	Enter an asterisk to clear all crossbar alarms on the active PXM45.
slot	The slot number of the crossbar planes. The slot number is 7 or 8.
plane	The range for plane numbers is 0–2.

Related Commands

dspxbaralm, **dspswalms**

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Clear the alarms for switch plane 0 in slot 7. The system returns no messages unless an error exists in the command syntax. To see the results, execute **dspxbaralm**.

```
pop20two.7.PXM.a > clrxbaralm 7 0
```

clrxbarerrcnt

Clear Crossbar Error Count—clear the counters for crossbar errors.

To see the errors that **clrxbarerrcnt** clears, execute **dspxbarerrcnt**. The **clrxbarerrcnt** command clears the errors for either a specific switch plane or all the switch planes on the active switching card. In the MGX 8850 node, the switching slot is the slot number of the PXM45.

Cards on Which This Command Runs

PXM45

Syntax

clrxbarerrcnt < * | [*slot plane*]>

Syntax Description

*	Enter an asterisk to clear all crossbar errors on the active PXM45.
slot	The slot number of the crossbar planes. The slot number is 7 or 8.
plane	The range for plane numbers is 0–2.

Related Commands

dspxbarerrcnt, **cnfxbarerrthresh**, **dspxbarerrthresh**, **dspswalms**

Attributes

Log: log	State: active, standby	Privilege: SUPER_GP

Example

Clear the errors for switch plane 0 in slot 7. The system returns no messages unless an error exists in the command syntax. To see the results, execute **dspxbarerrcnt**.

```
pop20two.7.PXM.a > clrxbarerrcnt 7 0
```

cnfalm

Configure Alarm

Configures statistical alarm thresholds for a line. The configurable items for SONET and PLCP are defined in RFC 2258. The configurable items for DS3 and E3 are defined in RFC 2496. The items that constitute a configuration are:

- Line type: SONET, DS3, E3, or PLCP
- Tested layer: section, line, or path (for example, SONET line)
- Test periods of 15 minutes and 24 hours
- Degrees of error-time: *errored seconds* and *severely errored seconds*
- Types of errors, including framing errors, code violations, and unavailable
- Severity of alarm triggered when a threshold is crossed: minor or major

A keyword identifies the alarm criteria. Each keyword identifies the tested layer (line, and so on), the type of threshold (errored seconds, and so on), and the test period of 15 minutes or 24 hours. For example, -**lnes15** indicates the number of errored seconds on the line layer during any 15 minute period. See the Syntax Description for a list and definitions of all keywords.

Cards on Which This Command Runs

AXSM

Syntax

The required parameters are the line type the line identifier in the format *bay.line*, and the severity of the alarm (minor or major). All other parameters are optional and must be preceded by the keyword that identifies the type of parameter.

Generic Syntax Description

The generic syntax is.

`cnfalm` `<`*`line type`*`> <`*`bay.line`*`> <`*`alarm severity`*`> <`*`thresholds`*`>`

The meaning of the generic syntax appears in the following list. Refer to subsequent lists for the descriptions of alarm severities and thresholds for each *line type*.

line type	The line type is specified as one of the following keywords (including the hyphen): -sonetsec (for SONET section) -sonetline (for SONET line) -sonetpath (for SONET path) -ds3 -e3 -plcp
bay.line	Identifies the bay (1 or 2) and the number of the line. The range for *line* can be 1 to the highest numbered line on the back card.

alarm severity	A keyword and number to identify the severity of the alarm that is triggered when any of the specified thresholds is crossed: 1 = minor alarm, and 2 = major alarm. Precede the alarm severity with the appropriate keyword. For the alarm severity keyword for each *line type*, see the first item in each of the lists follows. (For example, -**secsev** refers to the severity of the section alarm on a SONET line.)
thresholds	The number of instances of whatever the keyword identifies. The range for each *threshold* is 1 to 2^32-1. The keyword precedes each *threshold*. For example, **-lnsesf15 10** means 10 instances of severely errored framing seconds on a line during a 15-minute period.

Thresholds for SONET Section

-secsev <Severity>	Severity of the alarm (1 = minor, 2 = major) for SONET section.
-seces15 <ES15min>	Errored seconds during a 15-minute period.
-seces24 <ES24Hr>	Errored seconds during a 24-hour period.
-secses15 <SES15min>	Severely errored seconds during a 15-minute period.
-secses24 <SES24Hr>	Severely errored seconds during a 24-hour period.
-secsefs15 <SEFS15min>	Severely errored frame seconds during a 15-minute period.
-secsefs24 <SEFS24Hr>	Severely errored frame seconds during a 24-hour period.
-seccv15 <UAS15min>	Unavailable seconds during a 15-minute period.
-seccv24 <UAS24Hr>	Unavailable seconds during a 24-hour period.

Thresholds for SONET Line

-lnsev <Severity>	Severity of the alarm (1 = minor, 2 = major) for SONET line.
-lnes15 <ES15min>	Errored seconds during a 15-minute period.
-lnes24 <ES24Hr>	Errored seconds during a 24-hour period.
-lnses15 <SES15min>	Severely errored seconds during a 15-minute period.
-lnses24 <SES24Hr>	Severely errored seconds during a 24-hour period.
-lncv15 <CV15min>	Code violations during a 15-minute period.
-lncv24 <CV24Hr>	Code violations seconds during a 24-hour period.

-lnuas15 <UAS15min>	Unavailable seconds during a 15-minute period.
-lnuas24 <UAS24Hr>	Unavailable seconds during a 24-hour period.

Thresholds for SONET Path

-sev	Severity of the alarm (1 = minor, 2 = major) for SONET path.
-es15 <ES15min>	Errored seconds during a 15-minute period.
-es24 <ES24Hr>	Errored seconds during a 24-hour period.
-ses15 <SES15min>	Severely errored seconds during a 15-minute period.
-ses24 <SES24Hr>	Severely errored seconds during a 24-hour period.
-cv15 <CV15min>	Code violations during a 15-minute period.
-cv24 <CV24Hr>	Code violations seconds during a 24-hour period.
-uas15 <UAS15min>	Unavailable seconds during a 15-minute period.
-uas24 <UAS24Hr>	Unavailable seconds during a 24-hour period.

Thresholds for DS3

-dsev <severity>	Severity of the alarm (1 = minor, 2 = major) for DS3.
-lcv15 <LCV15min>	Code violations for a line during a 15-minute period.
-lcv24 <LCV24Hr>	Code violations for a line seconds during a 24-hour period.
-les15 <LES15min>	Line errored seconds during a 15-minute period.
-les24 <LES24Hr>	Line errored seconds during a 24-hour period.
-pcv15 <PCV15min>	P-bit coding violations for a line during a 15-minute period.
-pcv24 <PCV24Hr>	P-bit coding violations for a line during a 24-hour period.
-pes15 <PES15min>	P-bit errored seconds during a 15-minute period.
-pes24 <PES24Hr>	P-bit errored seconds during a 24-hour period.
-pses15 <PSES15min>	P-bit severely errored seconds during a 15-minute period.
-pses24 <PSES24Hr>	P-bit severely errored seconds during a 24-hour period.

-sefs15 <SEFS15min>	Severely errored frame seconds during a 15-minute period.
-sefs24 <SEFS24Hr>	Severely errored frame seconds during a 24-hour period.
-uas15 <UAS15min>	Unavailable seconds during a 15-minute period.
-uas24 <UAS24Hr>	Unavailable seconds during a 24-hour period.

Thresholds for E3

-dsev <severity>	Severity of the alarm (1 = minor, 2 = major) for DS3.
-lcv15 <LCV15min>	Code violations for a line during a 15-minute period.
-lcv24 <LCV24Hr>	Code violations for a line seconds during a 24-hour period.
-les15 <LES15min>	Line errored seconds during a 15-minute period.
-les24 <LES24Hr>	Line errored seconds during a 24-hour period.
-sefs15 <SEFS15min>	Severely errored frame seconds during a 15-minute period.
-sefs24 <SEFS24Hr>	Severely errored frame seconds during a 24-hour period.
-duas15 <UAS15min>	Unavailable seconds during a 15-minute period.
-duast24 <UAS24Hr>	Unavailable seconds during a 24-hour period.

Thresholds for PLCP

-psev <severity>	Severity of the alarm (1 = minor, 2 = major) for PLCP.
-bcv15 <CV15min>	Bipolar violation code violations during a 15-minute period.
-bcv24 <CV24Hr>	Bipolar violation code violations during a 24-hour period.
-bes15 <ES15min>	Bipolar violation errored seconds during a 15-minute period.
-bes24 <ES24Hr>	Bipolar violation errored seconds during a 24-hour period.
-bses15 <SES15min>	Bipolar violation severely errored seconds during a 15-minute period.
-bses24 <SES24Hr>	Bipolar violation severely errored seconds during a 24-hour period.
-psefs15 <SEFS15min>	PLCP severely errored frame seconds during a 15-minute period.
-psefs24 <SEFS24Hr>	PLCP severely errored frame seconds during a 24-hour period.

-puas15 <UAS15min>	PLCP unavailable seconds during a 15-minute period.
-puas24<UAS24Hr>	PLCP unavailable seconds during a 24-hour period.

Related Commands

dspalmcnf

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Configure the following thresholds for triggering a major line-level alarm on line 2 in bay 1:

- The *line type* is SONET line.
- The bay is 1, and the line number is 2.
- The severity of the triggered alarm is major.
- The errored seconds for a 15-minutes period and a 24-hour period are 60 and 600, respectively.
- The severely errored seconds for a 15-minutes period and a 24-hour period are 3 and 7, respectively.
- The code violations for a 15-minutes period and a 24-hour period are 75 and 750, respectively.
- The unavailable seconds for a 15-minutes period and a 24-hour period are 10 and 10, respectively

```
node4.1.AXSM.a > cnfalm -sonetline 1.2 -lnsev 2 -lnes15 60 -lnes24 600 -lnses15 3 -lnses24
7 -lncv15 75 -lncv24 750 -lnuas15 10 -lnuas24 10
```

Check the configuration by executing **dspalmcnf** for the line number and line type in this example.

```
node4.1.AXSM.a > dspalmcnf -sonetline 1.2
  LineNum: 1.2
  Line Stat Alarm Severity: No Alarm
               15min Threshold    24hr Threshold
  Line  ESs :  60                 600
  Line  SESs:  3                  7
  Line  CVs :  75                 750
  Line  UASs:  10                 10
```

cnfdiag

Configure Diagnostics

Enables the online or offline diagnostics. The **cnfdiag** command also configures the time settings for the start time and coverage for running the offline diagnostics. When you enter **cnfdiag** with no parameters, it displays the current configuration and status of the diagnostics.

The **cnfdiagall** command is the same as **cnfdiag** except that it configures all slots on the card at once.

The Purpose of the MGX 8850 Diagnostics

MGX 8850 diagnostics were implemented to test and validate the communication paths on the controller (PXM45) and the service modules (AXSM) to ensure reliability before and during operation. The diagnostics are always scheduled from the PXM45 controller card whether or not they run on the PXM45 card or the AXSM card.

For backward compatibility, the MGX 8850 Release 2.1 switch has two buses on its backplane:

- A 1.2 Gbps Cellbus
- A 45 Gbps MGX 8850 Release 2.1 bus

Because of the difference in bus speeds on the backplane, the Reliability Availability Serviceability (RAS) requirements demand that diagnostics be run periodically on the communications paths.

Consequently, diagnostics should be running periodically on both active and standby cards, but especially on standby cards. It is important that standby cards are tested using diagnostics periodically and frequently so that when an active card fails, the standby card has been tested and is ready to assume the active card state immediately.

Online Diagnostics

Online diagnostics are nondestructive tests (that do not interfere with active traffic) and run on either active and standby cards. The MGX 8850 switch supports seven online diagnostics tests. Three tests run on the PXM45 card, and four tests run on the AXSM card.

PXM45 Online Diagnostics

Active State

When you enable online diagnostics on an active PXM45 card, the following test runs:

- Crossbar loopback test on QE1210, Humvee, and Crossbar paths

Standby State

When you enable online diagnostics on a standby PXM45 card, the following tests run:

- Framer loopback test on QE1210, ATLAS, and OC-3 Framer paths
- Crossbar loopback walk test on QE1210, Humvee, and Crossbar paths

AXSM Online Diagnostics

Active State

When you enable online diagnostics on an active AXSM card, the following test runs:

- Crossbar loopback test on QE, Humvee, and Crossbar paths

Standby State

When you enable online diagnostics on a standby AXSM card, the following tests run:

- Crossbar loopback walk test on QE, Humvee, and Crossbar paths
- Back card loopback test on T3/E3 lines

Offline Diagnostics

Offline diagnostics are destructive tests (that interfere with active traffic) and therefore run *only* on standby cards.

Offline diagnostics must be scheduled using the offline start (*offStart*) and offline day-of-week (*offDow*) parameters. The coverage (*offCover*) parameter specifies the length of time that the offline diagnostics will run.

Note When an active card fails, the shelf manager must stop the diagnostics on the standby card immediately, reset, and allow normal arbitration to occur.

When offline diagnostics is enable and scheduled, numerous tests may be run depending on the implementation. The possible tests that may be run on the PXM45 and AXSM cards are listed in the following sections.

PXM45 Offline Diagnostics

1. Processor Subsystem Test
   - Flash EPROM
   - SDRAM
   - SCC
   - PCI Bridge
2. Component Level Test
   - PIO
   - FPGA
   - LEDs
   - Fan / Power Supply
   - BRAM / RTC
   - SEEPROM / NovRAM Checksum Test
3. ASIC Tests
   - QE1210 Register and DMA RAM Test
   - CBC Register Test
   - Humvee Register Test
   - Switch ASIC Register Test
   - Atlas Register and RAM Test
   - Framer Register and RAM Test
4. UI S3/S4 Back card test

- NovRAM Checksum Test

5. Cell Path Test
   - CBC Cell Path Test - Backplane Side
   - Framer Cell Path Test - Port Side
   - Humvee / Transceiver / Crossbar Switch Cell Path Test

AXSM Offline Diagnostics

1. I/O PLD data bus test
2. Skystone Framer register & RAM test.
3. Humvee ASIC register test
4. CBC register test
5. ATMizer RAM test
6. QE48 register and RAM test
7. NovRAM checksum test
8. Flash EPROM checksum test

Syntax

```
cnfdiag <slot> <onEnb> <offEnb>
[<offCover> <offStart> <offDow>]
```

Syntax Description

slot	The slot of the card for which to configure the diagnostics.
onEnb	Enable or disable online diagnostics. The default is disable.
offEnb	Enable or disable offline diagnostics. The default is disable.
offCover	Set the offline diagnostics coverage time to light, medium, or full. • light = 5 minutes or less • medium = 30 minutes or less • full = 60 minutes or more
offStart	Set the time for the offline diagnostics to begin using 24 hour time. The format is: hh:mm. For example: 03:45 or 22:30.
offDow	Sets the day of the week for the offline diagnostics to run.The format is SMTWTFS. For example: -M-W--- is Mondays and Wednesdays only

Cards on Which This Command Runs

PXM45

Related Commands

cnfdiagall, dspdiagcnf

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

Example

```
cnfdiag 7 enable disable light 22:30 -M-W-F-
```

cnfdiagall

Configure Diagnostics All

This command enables and configures online or offline diagnostics for all card slots. (This command is the same as **cnfdiag** except that it effects all slots instead of just one.)

When you enter this command with no parameters, it displays the current configuration and status of the diagnostics.

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

```
cnfdiagall <onEnb> <offEnb>
[<offCover> <offStart> <offDow>]
```

Syntax Description

onEnb	Enables online diagnostics.
offEnb	Enables offline diagnostics.
offCover	Sets the offline diagnostics coverage time to light, medium, or full. • light = 5 minutes or less • medium = 30 minutes or less • full = unlimited
offStart	Sets the time for the offline diagnostics to begin using 24 hour time. The format is: hh:mm For example: 03:45 or 22:30.
offDow	Sets the day of the week for the offline diagnostics to run. The format is SMTWTFS. Enter a dash (-) for days on which you do not want diagnostics to run. For example: -M-W--- is Mondays and Wednesdays only

Cards on Which This Command Runs

PXM45

Related Commands

cnfdiag, dspdiagcnf

Attributes

Log: no log	State: active, standby	Privilege: SERVICE_GP

Example

```
cnfdiagall enable disable light 22:30 -M-W-F-
```

cnfpnportloscallrel

Configure PNNI Port Loss of Signal Call Release

The **cnfpnportloscallrel** command lets you shut off the standard delay for rerouting calls on a port when the system detects loss of signal (LOS) on a port.

When the system detects LOS on an NNI link, the switch does not immediately tear down the calls on the link—in case the break is momentary. By default, the system waits for the SSCOP "no-response" and T309 timers to time out before it releases calls on the broken link. The default values for these timers are 30 seconds and 10 seconds, respectively. The system-level assumption (and therefore the default for **cnfpnportloscallrel**) is to retain all the calls for a temporary loss of connectivity, but this can also have the effect of delaying the rerouting of connections. The **cnfpnportloscallrel** command lets you direct the system to reroute calls without delay on a particular port.

Cards on Which This Command Runs

PXM45

Syntax

cnfpnportloscallrel <*portid*> <**yes | no**>

Syntax Description

portid	The *portid* represents the PNNI logical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.
yes | *no*	Specifies whether immediate call release is enabled upon LOS. To enable this feature—to remove the standard reroute delay—type "yes." Default: no.

Related Commands

dsppnportloscallrel

Attributes

Log: log	State: active	Privilege: SUPER_GP

Example

Enable call release upon LOS for port 3:1.1:1, then confirm its status.

8850_NY.8.PXM.a > **cnfpnportloscallrel** 3:1.1:1 yes

8850_NY.8.PXM.a > **dsppnportloscallrel** 3:1.1:1

```
Call release on Los :enabled
```

cnfxbarerrthresh

Configure Crossbar Error Threshold—specify a threshold for a particular crossbar error

The Syntax Description contains a list of possible errors. A crossbar error *threshold* consists of:

- A period for counting errors
- Severity of the resulting alarm (minor, major, and critical)
- Upper and lower counts for each alarm severity

Note The default settings for crossbar error thresholds are optimal for nearly all applications. The **dspxbarerrthresh** command shows the existing thresholds. If necessary, you can change thresholds through the **cnfxbarerrthresh** command.

Cards on Which This Command Runs

PXM45

Syntax

cnfxbarerrthresh <*errtype*(1..9)> <*threshtime*> <*severity*(0..2)> <*clrcount*> <*almcount*>

Syntax Description

errtype	A number that identifies the type of error, as follows: 1. Loss of synchronization (LossOfSync). 2. Transceiver error (TranscieverErr) 3. DisparityErr—an accumulation of five ASIC-level errors 4. ParityErr—a parity error in the switch frame as a whole 5. HeaderCRCErr—a CRC error for the switch frame header 6. PayloadCRCErr—a CRC error for the switch frame payload 7. RemapTwiceErr 8. RemapRecurrErr 9. Backpressure parity error (B.P.ParityErr)—a parity error in the signaling for backpressure
threshtime	The number of milliseconds over which the system counts errors.
severity	The severity of the alarm resulting from the error count per threshold time.
clrcnt	The clear count is the number of errors below which the alarm changes to the next lowest severity. For example, the system clears a minor alarm for a particular type of error when the number of errors goes to 0. Similarly, if *clrcnt* for a major alarm is 30, the alarm goes to minor when the count drops below 30.

almcnt	The number of errors for an alarm severity above which the alarm goes to the next highest severity.

Usage Guidelines for cnfxbarerrthresh

You can change the threshold for only one type of error at a time. Also, you must enter all parameters of the threshold whether or not you change them. For example, if you want to change only the duration of errored milliseconds, you must include the existing parameters. Therefore, you probably need to execute **dspxbarerrthresh** before **cnfxbarerrthresh**.

Related Commands

dspxbarerrthresh

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

For Loss of Sync, set the clear count for critical alarms to 200.

```
pop20two.7.PXM.a > cnfxbarerrthresh 1 2000 2 200 301
```

In the sequence of command and arguments, the only value that differs from the existing threshold is the clear count of 200. If the operation is successful, the system displays the error threshold for the type of error you specified—Loss of Sync in this example.

```
pop20two                        System Rev: 02.01   Dec. 05, 2000 02:29:17 GMT
MGX8850                                             Node Alarm: MAJOR
                           CROSSBAR ERROR CONFIGURATION
                     Thresh  -- MINOR --     -- MAJOR --     -- CRITICAL --
Device Error         Time    Clear Alarm     Clear Alarm      Clear Alarm
Type                 (msec)  Count Count     Count Count      Count Count
-------------------- ------  ----- -----     ----- -----      ----- -----
LossOfSync             2000      0     3         4    15        200   301
```

cnfxbarmgmt

Configure Crossbar Management—configure load sharing

The application for is a redundant PXM45 setup. It allows the MGX 8850 to maintain its peak throughput of 45 Gbps without requiring a switchover to the standby cards if a switch ASIC becomes defective. In this scheme, one of the switch ASICs on the standby PXM45 takes over the switching for the defective ASIC on the active PXM45.

Note On an MGX 8850 node, this command applies to Release 2.1 or higher.

Regardless of whether the node has redundant PXM45s or a load-sharing configuration, you can still investigate alarms and errors through a hierarchy of shelf-management and crossbar-related commands:

1. **dspndalms**
2. **dspswalms**
3. **dspxbaralm**
4. **dspxbarerrcnt**

Cards on Which This Command Runs

PXM45

Syntax

```
cnfxbarmgmt
<loadSharing>
<autoShutdown>
<planeAlarmThresh>
```

Syntax Description

loadSharing	0 = disable load sharing. 1 = enable load sharing. However, if a bad switch ASIC already exists, the switch blocks the command. –1 = force load sharing to be disabled when one or more bad switch ASICs exist on the active PXM45.
autoShutdown	A 0 disables automatic shut-down. A 1 enables automatic shut-down. The default is disabled.
planeAlarmThresh	An alarm threshold for declaring that a switch plane is bad. Each unit of the threshold represents a link between the switch ASIC and the card. (The determination of a bad link depends on the crossbar error threshold.) If the number of bad links reaches the threshold, the active PXM45 shuts down the ASIC and shifts the switching load to the standby PXM45. The range for *planeAlarmThresh* is 1–32.

Related Commands

dspxbarmgmt, **dspxbarerrthresh**, **dspxbarerrcnt**

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

Specify an error threshold of five links to declare a bad switch plane and enable auto-shutdown. The system returns a message only if an error occurs. Use **dspxbarmgmt** to check the new configuration.

```
pop20two.7.PXM.a > cnfxbarmgmt 1 1 5

pop20two.7.PXM.a > dspxbarmgmt
pop20two                         System Rev: 02.01   Dec. 06, 2000 00:44:20 GMT
MGX8850                                              Node Alarm: MAJOR
Load Sharing: Enable
Auto Shutdown: Enable
Plane Alarm Threshold: 5
```

conntrace

Call Control Operations

Trace an established connection and display the result.

Cards on Which This Command Runs

PXM45

Syntax

conntrace <*portid*>
callref
[*EndPtRef*]

Syntax Description

portid	The *portid* represents the PNNI logical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.
callref	Call reference for the call.
EndPtRef	Endpoint reference for a p2mp call. If no endpoint reference is specified, this is a p2p call.

Related Commands

pathtraceport, pathracenode, pathtraceie

Attributes

Log: log	State: active	Privilege: SUPER_GP

Examples

```
Result:Succ/Fail Reason: " Desc"
InterfaceId: "--" EndptRef: "--"
Originating Interface VPI: " --"
Originating Interface VCI: " --"
Originating Interface CallRef: " --"
NodeId Egress Port Vpi Vci CallRef PhysPortid
XXXX ZZZZ aaa bbb cccc eeee
XXXX ZZZZ aaa bbb cccc eeee
Terminating Interface VPI: " --"
Terminating Interface VCI: " --"
Terminating Interface CallRef: " --"
```

copycons

Copy Channels (Debugging Command)

Copies one or more endpoints from a specified endpoint. The number of entries to copy is specified using the *-num* option.

This command works by incrementing the VCI for a VCC endpoints and the VPI for a VPC endpoints.

The following steps are recommended when using this command:

1. First add a slave endpoint and then a master endpoint.
2. Copy the slave endpoints first using copychans.
3. Copy the master endpoints next using copychans.

Caution Improper use of this command can result in dangling (unpaired) endpoints in the network.

Cards on Which This Command Runs

AXSM

Syntax

```
copycons <source> <destn>
[-rem <remote Conn Id>]
[-num <num. conns to add>]
[-verbose <1|0>]
```

Syntax Description

source	source ID: The endpoint that is to be used as a template for copying, specified in the format: *ifNum.vpi.vci*
destn	destination ID: The endpoint to paste the copied connection template into, specified in the format: *ifNum.vpi.vci*
-num	The number of consecutive endpoints to be added, starting from *destn* endpoint. Default: 1
-rem	The remote connection ID specified in the format: *ifNum.vpi.vci*
-verbose	Prints the status of cloning process if enabled. Default is disabled. verbose: Enable(1) | Disable (0)

Related Commands

addcon, delcon

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

```
MGX8850.1.11.AXSM.a > copycons 3.10.50 3.10.60 -num 10
```

dbgcon

Debug Connection—enables or disables logging of SPVC-related errors.

Enable or disable the SPVC log.To see whether the SPVC log is enabled, use **dspcons-dbg**.

Cards on Which This Command Runs

PXM45

Syntax

```
dbgcon
<enable | disable>
```

Syntax Description

enable | disable	Enable or disable the SPVC log. Type the entire word. The default is disabled.

Related Commands

dspcons-dbg

Attributes

Log: no log	State: active, standby	Privilege: SERVICE_GP

Example

Enable the SPVC log.

```
MGX8850.8.PXM.a > dbgcon enable
```

delchanloop

Delete Channel Loopback

Delete a loopback from a connection (channel). For an understanding of the purpose of channel loopbacks, see the description of **addchanloop**.

Cards on Which This Command Runs

AXSM

Syntax

delchanloop *<ifNumber> <vpi> <vci>*

Syntax Description

ifNumber	The logical port number. The range for AXSM is 1–60.
vpi	The VPI of the connection. The range is 0–4095.
vci	The VCI of the connection. The range is 1–65535.

Related Commands

addchanloop, **dspchanloop**

Attributes

Log: log	State: active, standby	Privilege: SERVICE_GP

Example

Remove the loopback from VPI/VCI 1 50 on logical port 4.

```
pop20two.1.AXSM.a > delchanloop 4 1 50
```

dellnloop

Delete Line Loop

Remove the line loopback state from a line.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dellnloop <-**ds3** | -**sonet**> <*bay.line*>

Note For AXSM cards, the keyword **ds3** applies to both T3 and E3 line types.

Syntax Description

-**ds3** | -**sonet** *bay.line*	Specifies a SONET line (OC-3c, OC-12c, OC-48c) or a DS3 line (E3 or T3), the bay (1 for upper or 2 for lower), and the line number. The line number ranges from 1 to the highest number line on the back card.

Related Commands

addlnloop

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Deleting a DS3 loopback line.

```
MGX8850.1.11.AXSN.a > dellnloop -ds3 1.6 -lpb 3
Line loop-back status will be changed.
Do you want to proceed (Yes/No) ?
```

dncon

Down Connection

Temporarily deactivates (or downs) a connection so you can modify or troubleshoot it. The **dncon** command applies to only an SPVC. Execute at the master endpoint of a connection. Subsequently, when you view commands with **dspconinfo** or **dspcon** at the master endpoint, the display shows the connection state as down. If you execute either of these commands at the slave endpoint, the state appears as failed.

To reactivate the connection, use **upcon**.

Cards on Which This Command Runs

AXSM

Syntax

dncon *<ifNum> <vpi> <vci>*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	Virtual path identifier in the range 1–255 for a UNI or 1–4095 for an NNI.
vci	For a virtual connection (VCC), the VCI range is 0–65535. For a virtual path connection (VPC), the VCI is always 0.

Related Commands

upcon

Attributes

Log: log	State: active	Privilege: GROUP1

dnln

Down Line

Use **dnln** to de-activate a line on the current card. Before you can de-activate a line, you must:

Step 1 Remove connections. Use **delcon** or **delcons**.

Step 2 Remove any resource partitions. Use **dsprscprtn** to see existing partitions and **delrscprtn** to remove partitions.

Step 3 Remove all logical ports. Use **dspports** to see existing logical ports on the line and **delport** to remove logical ports.

Cards on Which This Command Runs

AXSM

Syntax

`dnln` *`<bay.line>`*

Syntax Description

bay.line	Specifies a SONET line (OC-3c, OC-12c, OC-48c) or a DS3 line (E3 or T3), the bay (1 for upper or 2 for lower), and the line number. The line number ranges from 1 to the highest number line on the back card.

Related Commands

dspln, **dsplns**, **cnfln**, **upln**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

De-activate line 1 in bay 1.

```
MGX8850.1.AXSM.a > dnln 1.1
```

dnport

Down Port

The **dnport** command disables (or downs) a logical port and thereby halts all traffic on the logical port. The usual purpose for using **dnport** is troubleshooting. The configuration for the port remains intact whether the logical port is a UNI or an NNI. The command for enabling a downed port is **upport**.

For an NNI, the PXM45 de-routes the failed connections then re-routes them through other trunks. After you re-enable an NNI port through **upport**, you cannot return the re-routed connections to the upped port. The PXM45 routes connections over the trunk as needed.

On a UNI, the connections continue to exist, but remain in the failed state until you enable the port by executing **upport**.

Cards on Which This Command Runs

AXSM

Syntax

dnport <*ifNum*>

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60. Use **dspports** or **dspport** as needed to determine the need to disable a port.

Related Commands

dspport, **dspports**, **upport**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Disable port 1 on the current card.

```
MGX8850.1.AXSM.a > dnport 1
```

dspalm

Display Alarm—displays the alarms from among the configured alarm types for a line.

Use the **dspalm** command to view the alarms associated with a specified line. See **cnfalm** for a description of the types of alarms you can see. In addition to the configurable alarm types, the output also shows instances of loss of cell delineation (LOCD).

Cards on Which This Command Runs

AXSM

AXSM Syntax

dspalm -ds3|-e3|-sonet|-ds1|-e1 *<bay.Line>*

Syntax Description

-ds3 | -e3 | -sonet -ds1 | -e1	Specifies a SONET line (OC-3c, OC-12c, OC-48c), DS3 line, E3 line. or DS1 line.
<bay>	The bay (1 for upper or 2 for lower), and the line number. The line number ranges from 1 to the highest number line on the back card.
<line>	Line number: • For OC12: 1 • For OC3: 1–4 • T3, E3: 1–8

Related Commands

cnfalm, **clralm**, **dspalms**, **dspalmcnt**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display alarms on OC-12 line 1.1. In this example, the components of a SONET line (section, line, and path) are clear. Also, no instances of loss of cell delineation (LOCD) have occurred.

```
MGX8850.1.AXSM.a > dspalm -sonet 1.1
 Line Number            : 1.1
 Section Alarm State    : LOS
 Line Alarm State       : Clear
 Path Alarm State       : Clear
 Section Stat Alarm State: TotalESs,TotalSESs,TotalSEFSs,CurrentESs,CurrentSESss
 Line Stat Alarm State  : TotalUASs,CurrentUASs
 Path Stat Alarm State  : TotalUASs,CurrentUASs
 LOCD Alarm State       : Clear
 APS  Alarm State       : N/A
```

On another node, the same bay.line shows some of the possible errors: loss of signal (LOS), errored seconds and severely errored seconds, unavailable seconds.

```
MGX8850.1.AXSM.a > dspalm -sonet 1.1
 Line Number            : 1.1
 Section Alarm State    : LOS
 Line Alarm State       : Clear
 Path Alarm State       : Clear
 Section Stat Alarm State: CurrentESs,CurrentSESs,CurrentSEFSs
 Line Stat Alarm State  : CurrentSESs,CurrentUASs
 Path Stat Alarm State  : CurrentSESs,CurrentUASs
 LOCD Alarm State       : Clear
 APS  Alarm State       : N/A
```

dspalmcnf

Display Alarm Configuration

Display the threshold information about the alarm statistics being collected. Refer to the **cnfalm** description for details regarding alarm threshold configuration.

Cards on Which This Command Runs

AXSM

Syntax

```
dspalmcnf
-sonetsec|-sonetline|-sonetpath|-ds3|-e3|-plcp <bay.line>
```

Syntax Description

-sonetsec **-sonetline** **-sonetpath** **-ds3-** **-e3** **-plcp**	Keywords that specify the type of alarms to display: • **-sonetsec** – Displays the section alarms for the given sonet line (*bay.line*). • **sonetline** – Displays the line alarms for the given sonet line (*bay.line*). • **-sonetpath** –Displays the path alarms for the given sonet line (*bay.line*). • **-ds3** – Displays the statistical alarms for the given DS3 line (*bay.line*). • **-e3** – Displays the statistical alarms for the given E3 line (*bay.line*). • **-plcp** – Displays the PLCP alarms for the given line (*bay.line*). Physical Layer Convergence Procedure (PLCP) is the specification that maps ATM cells into physical media, such as T3 or E3, and defines certain management information.
bay.line	Specifies the type of line, the bay (1 or 2), and the number of the line. The line number can be 1 to the highest numbered line on the back card.

Related Commands

cnfalm, **dspalm**, **dspalms**

Attributes

Log: no log	State: active, standby,	Privilege: ANYUSER

Examples

Display alarm configuration for AXSM cards.

```
MGX8850.1.AXSM.a > dspalmcnf -sonetline 1.2
  LineNum: 1.2
  Line Stat Alarm Severity: No Alarm
               15min Threshold    24hr Threshold
  Line  ESs :  60                 600
  Line  SESs:  3                  7
  Line  CVs :  75                 750
  Line  UASs:  10                 10

MGX8850.1.AXSM.a > dspalmcnf -sonetsec 1.1
  LineNum: 1.1
  Section Stat Alarm Severity: No Alarm
                   15min Threshold    24hr Threshold
  Section ESs  :   60                 600
  Section SESs :   3                  7
  Section SEFSs:   3                  7
  Section CVs  :   75                 750
```

dspalmcnt

Display Alarm Counters

Displays the performance monitoring alarm counters for either a SONET or DS3 line.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dspalmcnf -ds3|e3|-plcp|-sonet|ds1|-e1 <*bay.line*>

Syntax Description

-ds3 **e3** **-plcp** **-sonet** **ds1** **-e1**	Keywords that specify the type of alarm counters (see **dspalmcnf** for definitions) to display for the given line (*bay.line*) on an AXSM card.
bay.line	Specifies the type of line, the bay (1 or 2), and the number of the line. The line number can be 1 to the highest numbered line on the back card.

Related Commands

clralmcnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display the alarm count for T3 line 1 in bay 1.

```
MGX8850.11.AXSM.a > dspalmcnt -ds3 1.1
  Line Num:      2.1
  CurrentLCV :      9109365
  CurrentLES :      13
  CurrentPCV :      1
  CurrentPES :      1
  CurrentPSES:      0
  CurrentSEFS:      11
  CurrentUAS :      0
  Num of LOS :      1
  Num of OOF :      1
  Num of RAI :      0
  Num of CCV :      0
  Num of FE :       0
```

Display SONET line 1 in bay 1.

```
MGX8850.6.AXSM.a > dspalmcnt -sonet 1.1
 Line Num:                        1.1
 Elapsed Time (in sec):             1634
Section PM:
-----------
  Num of LOSs:                  1
  Num of LOFs:                  1
  CurrentESs:                   0
  CurrentSESs:                  0
  CurrentSEFSs:                 1
  CurrentCVs:                   1
Line PM:
--------
  Num of AISs:         0
  Num of RFIs:         0
  Near End                            Far End
  CurrentESs :              1           CurrentESs : 1
  CurrentSESs:              0           CurrentSESs: 0
  CurrentCVs :              1           CurrentCVs : 1
  CurrentUASs:              0           CurrentUASs: 0
Path PM:
--------
  Num of AISs:                      1
  Num of RFIs:                      1
  Near End                            Far End
  CurrentESs :              0           CurrentESs : 0
  CurrentSESs:              0           CurrentSESs: 0
  CurrentCVs :              0           CurrentCVs:  0
  CurrentUASs:              0           CurrentUASs: 0
```

dspalms

Display Alarms

Display all line-related alarms on the card. RFC 2258 describes the alarm categories. The display can easily scroll for many pages if more than one line is active. See **cnfalm** for a description of types of alarms you might see. In addition to the alarms from **cnfalm**, the **dspalms** command also displays instances of loss of cell delineation (LOCD).

Cards on Which This Command Runs

AXSM

Syntax

`dspalms`

Syntax Description

This command takes no parameters.

Related Commands

dspalm, clralm

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display alarms for the lines on the current AXSM card.

```
MGX8850.1.AXSM.a > dspalms
   Line Number: 1.1
   Alarm State
      Section : LOS
      Line    : Clear
      Path    : Clear
   Statistical Alarm State
      Section : TotalESs,TotalSESs,TotalSEFSs,CurrentESs,CurrentSESs,CurrentSEFSs
      Line    : TotalSESs,TotalUASs,CurrentUASs
      Path    : TotalSESs,TotalUASs,CurrentUASs
   LOCD Alarm : Clear

   Line Number: 1.2
   Alarm State
      Section : LOS
      Line    : Clear
      Path    : Clear
   Statistical Alarm State
      Section : TotalSESs,TotalSEFSs,CurrentESs,CurrentSESs,CurrentSEFSs
      Line    : TotalSESs,TotalUASs,CurrentUASs
      Path    : TotalSESs,TotalUASs,CurrentUASs
   LOCD Alarm : Clear
Line Number: 2.1
```

```
Alarm State
   Section : Clear
   Line    : Clear
   Path    : Clear
Statistical Alarm State
   Section : Clear
   Line    : Clear
   Path    : Clear
LOCD Alarm : Clear

Line Number: 2.2
Alarm State
   Section : Clear
   Line    : Clear
   Path    : Clear
Statistical Alarm State
   Section : Clear
   Line    : Clear
   Path    : Clear
LOCD Alarm : Clear
```

dspbecnt

Display Bit Error Count

The **dspbecnt** command lets you display the APS-related bit error counters.

Cards on Which This Command Runs

AXSM

Syntax

`dspbecnt <working-bay.line>`

Syntax Description

working-bay.line	Identifies the bay (1 or 2) and the number of the line. The line number is from 1 to the highest numbered line on the back card. For the range of line numbers on specific AXSM models, see Table 9-1.

Related Commands

addapsln, cnfapsln, delapsln, dspapsln, dspapslns, switchapsln, dspapsbkplane, dspbecnt

Attributes

Log: no log	State: active	Privilege: ANYUSER

dspcd

Display Card

Display the following information about a card:

- Hardware serial number.
- Firmware revision level. (See the **loadrev** description for an explanation of how to interpret the revision filed.) If a card has no firmware, the display shows a version number of 0.0.0.
- Status, possibly including the reason for the last reset (FunctionModuleResetReason) and state of the integrated alarm (cardIntegratedAlarm).
- For a service module only, a count of configured lines, ports, and connections.

Note The connection count includes control VCs when you execute **dspcd** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

- For a service module only: which physical lines constitute a *port group* and the maximum number of connections in that port group. A port group consists of one to many physical lines. This maximum connection count is a function of the hardware interface type (OC-3, OC-12, and so on). The port group information also shows the number of existing SVCs, SPVCs, and SPVPs.

  Use the maximum number of supported connections to help you configure resource partitions. If a particular resource partition has close to the maximum supported by hardware on a line, few or no connections would be possible in another partition on the same line.

Some of the information that **dspcd** shows is common to the **version** command, but **version** shows the boot code version in bold.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

```
dspcd
```

Syntax Description

This command does not take parameters.

Related Commands

dspcds, **version**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display card details for the current PXM45.

Note The A1 at the end of the primary software revision and boot firmware revision numbers shows that these versions are pre-release. Refer to the **setrev** description for details.

```
MGX8850.7.PXM.a > dspcd
MGX8850                          System Rev: 02.00   Aug. 02, 2000 23:39:06 GMT
MGX8850                                              Node Alarm: CRITICAL
Slot Number    7    Redundant Slot:  8

                    Front Card          Upper Card          Lower Card
                    ----------          ----------          ----------

Inserted Card:      PXM45               UI Stratum3         PXM HardDiskDrive
Reserved Card:      PXM45               UI Stratum3         PXM HardDiskDrive
State:              Active              Active              Active
Serial Number:      SAK0405008B         SAK0325009M         12345678901
Prim SW Rev:        2.0(246)A1          ---                 ---
Sec SW Rev:         2.0(246)A1          ---                 ---
Cur SW Rev:         2.0(246)A1          ---                 ---
Boot FW Rev:        2.0(168)A1          ---                 ---
800-level Rev:      18                  03                  22
Orderable Part#:    800-06147-01        800-05787-01        800-12345-99
CLEI Code:          0000000000          È                   1234567898
Reset Reason:       On Reset From Shell
Card Alarm:         NONE
Failed Reason:      None
Miscellaneous Information:

MGX8850                          System Rev: 02.00   Aug. 02, 2000 23:39:06 GMT
MGX8850                                              Node Alarm: CRITICAL

Crossbar Slot Status:      Present

Alarm Causes
------------
    NO ALARMS
```

Display card details for the current AXSM-1-2488.

```
MGX8850.1.AXSM.a > dspcd
                    Front Card          Upper Card          Lower Card
                    ----------          ----------        ------------

Card Type:          AXSM-1-2488         SMFSR-1-2488        ---

State:              Active              Present             Undefined

Serial Number:      SAK04010033         SAK040400F9         ---

Boot FW Rev:        2.0(233)A1          ---                 ---

SW Rev:             2.0(22)D            ---                 ---

800-level Rev:      09                  05                  ---

Orderable Part#:    800-5795-1          800-5490-2          ---

PCA Part#:          73-4363-1           73-4040-2           ---
```

```
Reset Reason:On Power up

Card SCT Id: 2

#Lines #Ports #Partitions   #SPVC    #SVC
------ ------ ----------- ------- -------
     1      1           1      2      1

Port Group[1]:
#Chans supported:32512  Lines:1.1
Port Group[2]:
#Chans supported:32512  Lines:1.2
Port Group[3]:
#Chans supported:32512  Lines:2.1
Port Group[4]:
#Chans supported:32512  Lines:2.2
```

dspcdalms

Display Card Alarms

Use **dspcdalms** on the PXM45 to display alarms that have been reported by a service module. If **dspcdalms** shows an alarm for one of the following parts of a card, you can **cc** to that card and execute one of the applicable commands:

- Line (**dsplns** and **dspln**)
- Port (**dspports** and **dspport**)
- Feeder (**dspfdr**, **dspfdrs**, and **dspfdrstat**)
- Connection (**dspcons** and **dspcon**)

In addition to the preceding, on the AXSM you can also execute **dspalm** and **dspalms**.

The definition of each alarm severity comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a disconnected line is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

The **dspcdalms** command is part of a hierarchy of troubleshooting commands you can execute on the PXM45 or a service module. Frequently, **dspcdalms** follows the higher-level command **dspndalms**.

Cards on Which This Command Runs

PXM45

Syntax

```
dspcdalms [slot]
```

Syntax Description

slot identifies a particular slot. For the current PXM45, *slot* is unnecessary. For any other card, you must include *slot.*

Related Commands

PXM45: **dspndalms**, **dspslotalms**, **dspswalms**, **dspclkalms**

AXSM: **dspalm**, **dspalms**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Display card-level alarms for the card in slot 8.

```
node19.8.PXM.a > dspcdalms 8
Node Card Alarm Summary

Line Alarm       Slot   8  Critical    0   Major    0   Minor    0
Port Alarm       Slot   8  Critical    0   Major    0   Minor    0
Connect Alarm    Slot   8  Critical    0   Major    0   Minor    0
```

Display card alarms without specifying a slot.

```
MGX8850.7.PXM.a > dspcdalms
Node Card Alarm Summary

Line Alarm       Slot   1  Critical    0   Major    0   Minor    0
Port Alarm       Slot   1  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   1  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   1  Critical    0   Major    0   Minor    0
Line Alarm       Slot   2  Critical    0   Major    0   Minor    0
Port Alarm       Slot   2  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   2  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   2  Critical    0   Major    0   Minor    0
Line Alarm       Slot   3  Critical    0   Major    0   Minor    0
Port Alarm       Slot   3  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   3  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   3  Critical    0   Major    2   Minor    0
Line Alarm       Slot   5  Critical    0   Major    0   Minor    0
Port Alarm       Slot   5  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   5  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   5  Critical    0   Major    0   Minor    0
Line Alarm       Slot   6  Critical    0   Major    0   Minor    0
Port Alarm       Slot   6  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   6  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   6  Critical    0   Major    0   Minor    0
Line Alarm       Slot   9  Critical    1   Major    0   Minor    0
Port Alarm       Slot   9  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot   9  Critical    0   Major    0   Minor    0
Channel Alarm    Slot   9  Critical    0   Major    0   Minor    0
Line Alarm       Slot  10  Critical    2   Major    0   Minor    1
Port Alarm       Slot  10  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot  10  Critical    0   Major    0   Minor    0
Channel Alarm    Slot  10  Critical    0   Major    0   Minor    0
Line Alarm       Slot  12  Critical    0   Major    0   Minor    0
Port Alarm       Slot  12  Critical    0   Major    0   Minor    0
Feeder Alarm     Slot  12  Critical    0   Major    0   Minor    0
Channel Alarm    Slot  12  Critical    0   Major    0   Minor    0


MGX8850.7.PXM.a > dspcdalms
Card Alarm Summary
Slot 1 Crit 0 Maj 0 Min 0 || Slot 17 Crit 0 Maj 0 Min 0
Slot 2 Crit 0 Maj 0 Min 0 || Slot 18 Crit 0 Maj 0 Min 0
Slot 3 Crit 0 Maj 0 Min 0 || Slot 19 Crit 0 Maj 0 Min 0
Slot 4 Crit 0 Maj 0 Min 0 || Slot 20 Crit 0 Maj 0 Min 0
Slot 5 Crit 0 Maj 0 Min 0 || Slot 21 Crit 0 Maj 0 Min 0
Slot 6 Crit 12 Maj 15 Min 19 || Slot 22 Crit 0 Maj 0 Min 0
Slot 7 Crit 0 Maj 0 Min 1 || Slot 23 Crit 0 Maj 0 Min 0
Slot 8 Crit 0 Maj 2 Min 0 || Slot 24 Crit 0 Maj 0 Min 0
Slot 9 Crit 0 Maj 0 Min 0 || Slot 25 Crit 0 Maj 0 Min 0
Slot 10 Crit 0 Maj 0 Min 0 || Slot 26 Crit 0 Maj 0 Min 0
Slot 11 Crit 0 Maj 0 Min 0 || Slot 27 Crit 0 Maj 0 Min 0
Slot 12 Crit 0 Maj 0 Min 0 || Slot 28 Crit 0 Maj 0 Min 0
```

```
Slot 13 Crit 0 Maj 0 Min 0 || Slot 29 Crit 0 Maj 0 Min 0
Slot 14 Crit 0 Maj 0 Min 0 || Slot 30 Crit 0 Maj 0 Min 0
Slot 15 Crit 0 Maj 0 Min 0 || Slot 31 Crit 0 Maj 0 Min 0
Slot 16 Crit 0 Maj 0 Min 0 || Slot 32 Crit 0 Maj 0 Min 0
Use dspcdalms <slot> to see more detail.

MGX8850.7.PXM.a > dspcdalms 6
Card Alarm Summary
Hardware Alarm Critical 0 Major 0 Minor 0
Card State Alarm Critical 0 Major 0 Minor 0
Disk Alarm Critical 0 Major 0 Minor 0
Line Alarm Critical 3 Major 4 Minor 5
Port Alarm Critical 4 Major 5 Minor 6
Feeder Alarm Critical 0 Major 0 Minor 0
Channel Alarm Critical 5 Major 6 Minor 7

MGX8850.7.PXM.a > dspndalms
Node Alarm Summary
Alarm Type Critical Major Minor
Clock Alarms 0 0 0
Switching Alarms 0 0 2
Environment Alarms 0 0 0

Card Alarms 12 17 18
```

dspcdbucketcnt

Display Cell Counts for the Card

The **dspcdbucketcnt** command shows the following cell-related counts:

- Cells transferred between the card and the backplane
- Cells from the QE 48
- CLP0 and CLP1 cells that the card dropped
- Invalid, errored, and unsupported OAM cells
- Errored RM cells

In addition to the other bucket command on the AXSM (**dsplnbucketcnt**), the display commands for the switch planes on the PXM45 may help you analyze cell flows. (See the **dspxbar**-type commands.)

Cards on Which This Command Runs

AXSM

Syntax

dspcdbucketcnt

Syntax Description

This command takes no parameters.

Related Commands

dsplnbucketcnt, all the **dspxbar**-type of commands except **dspxbarstatus**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the bucket counters for the current AXSM.

```
MGX8850.12.AXSM.a > dspcdbucketcnt
          cells to backplane(QLSI) : 0
          cells from QE 48 : 5347
          cells from backplane(QLSI) : 6917
          CLP0 cells dropped : 0
          CLP1 cells dropped : 0
          undefined cells from port : 0
          errored OAM from port : 0
          invalid OAM from port : 0
          unsupported OAM from port : 0
          errored RM cells from port :0
```

dspcdcnt

Display Card Counters

Displays the number of cells transferred between the service module and the switching planes. (Synonyms for switching plane are crossbar, xbar, and switch fabric.) One switch fabric is implemented in hardware by one ASIC. The **dspcdcnt** command primarily applies to debugging.

The type of information consists of:

- Cells transferred between the service module and each of the switch planes within the total array of switch planes.
- Total cells transferred between the service module and the backplane.
- Cells to and from QE48.
- Undefined cells.
- Total number of CLP0 and CLP1 cells that have been discarded.
- Errored, invalid, and unsupported OAM cells.
- Errored RM cells.
- Cells transferred between and individual switch plane and each slot. This information is centered on the switch fabric itself rather than the card. Each switch fabric can route cells to and from any slot, so the display includes this information for each switching plane.

Cards on Which This Command Runs

AXSM

Syntax

`dspcdcnt`

Related Commands

clrcdcnt (on the AXSM), **dspxbar**, **dspxbaralm**, **dspxbarerrcnt**, **dspxbarerrthresh**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Example

Display the cell transfers between the current AXSM and the switch planes (crossbar planes).

```
SanJose.4.AXSM.a > dspcdcnt
Ingress Count                          Egress Count
-------------                          ------------
Cells to xbar plane[1]:          1505  Cells from xbar plane[1]:  743
Cells to xbar plane[2]:             0  Cells from xbar plane[2]:  1
Cells to xbar plane[3]:          1488  Cells from xbar plane[3]:  445
Cells to xbar plane[4]:             0  Cells from xbar plane[4]:  0
Cells to xbar plane[5]:             0  Cells from xbar plane[5]:  0
Cells to xbar plane[6]:             0  Cells from xbar plane[6]:  0
Cells to xbar plane[7]:             0  Cells from xbar plane[7]:  0
Cells to xbar plane[8]:             0  Cells from xbar plane[8]:  0
Total cells to backplane:        2993  Total cells from backplane: 1189
Cells from QE48         :        2993  Cells to QE48:              1188
Undefined cells         :           0  CLP0 cells discard:         0
Errored OAM cells       :           0  CLP1 cells discard:         0
Invalid OAM cells       :           0
Unsupported OAM cells   :           0
Errored RM cells        :           0

Cells to dest slot[01]:             0  Cells to dest slot[02]:    0
Cells to dest slot[03]:             0  Cells to dest slot[04]:    0
Cells to dest slot[05]:             0  Cells to dest slot[06]:    0
Cells to dest slot[07]:          2993  Cells to dest slot[08]:    0
Cells to dest slot[09]:             0  Cells to dest slot[10]:    0
Cells to dest slot[11]:             0  Cells to dest slot[12]:    0
Cells to dest slot[13]:             0  Cells to dest slot[14]:    0
```

dspcderrs

Display Card Errors

Display information about card errors.

Cards on Which This Command Runs

PXM45

Syntax

`dspcderrs`

Related Commands

clrerr

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display all card errors.

```
wilco.7.PXM.a > dspcderrs

dspcderrs
08/05/95-18:53:05 tRootTask    3 Task failed          : scm
09/05/95-09:14:08 tRootTask    3 Task failed          : scm
value = 0 = 0x0

wilco.7.PXM.a >
```

dspcds

Display Cards

Displays high-level information for all the cards in the node. For more detailed information about a card, execute **dspcd** on the CLI of that card. The information that **dspcds** provides is the:

- Revision level of the boot firmware
- Revision level of the system software
- Date and time of command execution, including GMT offset
- Backplane serial number and its hardware revision level
- The IP address of the statistics master (a workstation)
- Type of card in the front and back slots and the (active/standby) state of each
- Alarm status for each card and the shelf itself
- Redundancy configuration for each slot

Cards on Which This Command Runs

PXM45

Syntax

dspcds

Related Commands

dspcd, **version**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display information for all cards in the MGX 8850 switch.

```
MGX8850.7.PXM.a > dspcds
MGX8850                          System Rev: 02.00   Aug. 06, 2000 18:03:35 GMT
Backplane Serial No: SAA03270618 Bp HW Rev:  B0      GMT Offset: 0
                                                     Node Alarm: CRITICAL
Card  Front/Back       Card           Alarm      Redundant    Redundancy
Slot  Card State       Type           Status     Slot         Type
---   ----------       --------       --------   -------      -----

01    Active/Active    AXSM_1OC48     NONE       NA           NO REDUNDANCY
02    Active/Active    AXSM_1OC48     NONE       NA           NO REDUNDANCY
03    Active/Active    AXSM_16OC3     NONE       04           PRIMARY SLOT
04    Standby/Active   AXSM_16OC3     NONE       03           SECONDARY SLOT
05    Active/Active    AXSM_4OC12     NONE       NA           NO REDUNDANCY
06    Active/Active    AXSM_4OC12     NONE       NA           NO REDUNDANCY
07    Active/Active    PXM45          NONE       08           PRIMARY SLOT
08    Standby/Active   PXM45          NONE       07           SECONDARY SLOT
09    Active/Active    AXSM_16T3E3    NONE       NA           NO REDUNDANCY
10    Active/Active    AXSM_16T3E3    NONE       NA           NO REDUNDANCY
11    Empty            ---            ---        ---          ---
12    Active/Active    AXSM_2OC12     NONE       NA           NO REDUNDANCY
13    Empty            ---            ---        ---          ---
14    Empty            ---            ---        ---          ---
```

dspcdstatus

Display Card Status

Displays the most serious alarms reported by a service module. The alarm information pertains to:

- Lines
- Ports
- Connections
- Feeders
- Severity of each alarm

You can use this command to isolate the alarm source if, for example, you see that a Critical Alarm LED is lit or just want to check the node for alarms. You can subsequently use other alarm commands to locate the problem. Some alarm commands run on only the PXM45, and other commands run on service modules. The commands other than **dspcdstatus** on the PXM45 are:

- **dspndalms** displays various types of alarms on the node from a high-level perspective. With the information in the **dspndalms** display, you can select one of the other commands to investigate the alarm further.
- **dspcdalms** identifies line, port, feeder, or connection alarms on an AXSM.
- **dspclkalms** shows alarms related to network clocks.
- **dspenvalms** lists alarms for out-of-range conditions for temperature, voltage sources, and so on.
- **dspslotalms** shows card-level alarms, such as a missing card or a disk problem on the PXM-HD.
- **dspswalms** shows alarms related to the switching hardware on the PXM45.

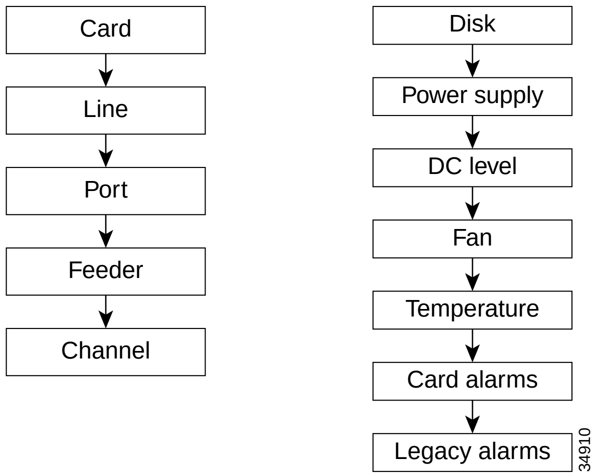
The alarm monitoring function on the PXM45 uses two criteria to determine which alarm to display. One criterion is alarm *severity*, and the other is *hierarchy*.

The definition of each *alarm severity* comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a line being disconnected is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

Two hierarchies of alarm types exist. They are card alarms and node alarms. See Figure 9-2 for a list of alarm categories. Note that, although the card alarms appear to apply to only service modules, this category can also apply to the PXM45.

Figure 9-2 Alarm Type Hierarchy



The alarm monitoring function reports the highest status alarm after it sorts the current alarms first by severity then by hierarchy. If alarms of equal severity exit in both hierarchies, the system reports the node alarm as the highest status alarm. For example, if a major alarm exists on a line and a major power alarm exists, the **dspcdstatus** command displays the power alarm as the highest status alarm.

Cards on Which This Command Runs

PXM45

Syntax

dspcdstatus

Syntax Description

This command requires a slot number.

Related Commands

dspndalms, **dspcdalms**, **dspenvalms**, **dspclkalms**, **dspswalms**, **dspalm** (AXSM), **dspalms** (AXSM)

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Display status of slot 11. The display shows a critical line alarm for slot 11. Next, cc to slot 11.

```
Golden_U2.8.PXM.a > dspcdstatus 11

Logical Slot  11    Physical Slot  11

Card Alarm Status - Type LINE              Severity CRITICAL
```

The display for **dspalms** shows that line 1 in bay 1 has LOS. (This example shows only the alarmed line. A complete display shows the status of all lines)

```
Golden_U2.11.AXSM.a > dspalms
   Line Number: 1.1
   Alarm State
      Section : LOS,LOF
      Line    : AIS
      Path    : RDI
```

dspchancnt

Display Channel Counters

Display the statistical counters for a connection (channel). See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Note This command does not apply to OC-48 cards.

Cards on Which This Command Runs

AXSM

Syntax

dspchancnt *<ifNum> <vpi> <vci> <isPVC>*

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	The VPI in the range 1–4095.
vci	The VCI in the range 1–65535.
isPVC	A Boolean expression that identifies either an SVC or a SPVC. Type a 0 for an SVC or a 1 for an SPVC.

Related Commands

dspchstats, clrchancnt, dspcdcnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display channel counters on AXSM for 1 10 100.

```
Golden_U2.11.AXSM.a > dspchancnt 1 10 100
                         Ingress         Egress
Instantaneous Qdepth:          0              0
Arrival CLP0 cells:            0         492305
Arrival CLP1 cells:            0              0
Dscd CLP0 cells:               0              -
Dscd CLP0+1 cells:             0              -
Noncompliant cells:            0              -
Arrival EFCI cells:            0              0
Arrival EOF cells:             0              0
```

dspchanloop

Display Channel Loopbacks

Show channel (connection) loopbacks on a logical port.

Cards on Which This Command Runs

AXSM

Syntax

`dspchanloop` *`<ifNumber>`*

Syntax Description

ifNumber	The logical port number. The range for AXSM is 1–60.

Related Commands

addchanloop, **delchanloop**

Attributes

Log: no log	State: active, standby	Privilege: SERVICE_GP

Example

Display any channel loopbacks on logical port 4. The display shows one connection with a loopback in the ingress direction.

```
pop20two.1.AXSM.a > dspchanloop 4
Port     Type      lVPI      lVCI      rVPI      rVCI
  4     igrLpbk       1        50         0        35
```

dspchantests

Display Channel Tests—display results of tstdelay or tstconseg commands.

The **tstdelay** or **tstconseg** commands test the integrity of the path of a connection in the ingress and egress directions, respectively. After you successfully start a test through **tstdelay** or **tstconseg**, the returned message directs you to use **dspchantests** or **dspcon** to see the results. The same test results presented by **dspchantests** appears in the **dspcon** display, but **dspchantests** shows only the test results.

Cards on Which This Command Runs

AXSM

Syntax

```
dspchantests <ifNum> <vpi> <vci>
[-num <count>]
```

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	The VPI range for the SVC or SPVC is 1–255.
vci	The VCI range for a VCC SPVC is 32–65535. For a VPC, the only VCI value for an SPVC is 0.
-num	(Optional) A keyword that indicates an aggregate connection count follows.
number	The number of connections to display.

Related Commands

tstdelay, **tstconseg**, **dspcon**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Run **tstdelay** on connection 1 10 100 then display the results.

Step 1 Execute **tstdelay**:

```
node19.1.AXSM.a > tstdelay 1 10 100
Test started; Use dspcon/dspchantests to see test results
```

Step 2 Check the results:

```
node19.1.AXSM.a > dspchantests 1 10 100
Connection Id    Test Type    Direction    Result     Round Trip Delay
=============    =========    =========    =======    ================
01.0010.00100:    OAM Lpbk     ingress     Success          40000
```

Run **tstconseg** for 1 10 100 then display the results.

Step 1 Run the test for 1 10 100:

```
node19.1.AXSM.a > tstconseg 1 10 100
Test started; Use dspcon/dspchantests to see test results
```

Step 2 Check the results:

```
node19.1.AXSM.a > dspchantests 1 10 100
Connection Id    Test Type    Direction    Result     Round Trip Delay
=============    =========    =========    =======    ================
01.0010.00100:    OAM Lpbk     egress      TimeOut              0
```

dspclkalms

Display Clock Alarms

Displays alarms associated with the primary or secondary clock source.

The switch constantly monitors the state of the clocks. On the local node, the clock monitor declares an alarm if the clock becomes undetectable or goes out of specification for any reason. The definition of each alarm severity comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a line being disconnected is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

The **dspclkalms** command is part of a hierarchy of troubleshooting commands you can execute on the PXM45 or a service module. Frequently, **dspclkalms** follows the higher-level command **dspndalms**. The **dspndalms** command shows a variety of alarms within the switch and helps isolate the problem.

The **dspndalms** and **dspclkalms** commands run on the PXM45. If the errored clock source appears to be on a service module, you can **cc** to the CLI of that card and execute a variety of alarm commands and other troubleshooting commands.

Cards on Which This Command Runs

PXM45

Syntax

```
dspclkalms
```

Syntax Description

This command takes no parameters.

Related Commands

dspcdstatus, **dspndalms**, **dspalm**, **dspalms**, **dspclksrcs**, **cnfclksrc**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display clock alarms.

```
pop20two.7.PXM.a > dspclkalms
pop20two                         System Rev: 02.00   Jul. 31, 2000 11:23:17 GMT
MGX8850                                              Shelf Alarm: NONE
Clock Manager Alarm Summary
----------------------------
Critical       Major         Minor
000            000           000
```

dspclksrcs

Display Clock Sources

Displays the configuration and status of the clock sources on the node. (For details about network synchronization, see the description of **cnfclksrc**.) The **dspclksrcs** output consists of:

- For the primary clock: the type, source, status, and reason (for status change) of the clock.
- For the secondary clock: the type, source, status, and reason (for status change) of the clock.
- The active clock—the clock that currently provides synchronization. The active clock can be primary, secondary, holdover, or internal.
- Whether revertive mode is enabled or disabled.

Note Changes to the configuration and status of clocks go into the database on the active PXM45. If a standby (redundant) PXM45 exists, it receives the initial clock configuration and status but receives internal status updates only when you interact with the node in a way that changes a configuration or when the standby PXM45 switches to the active state.

Type of Clock Source

The type is either BITS or generic. Currently, generic applies to only an AXSM-sourced clock. If a user-specified priority of clock is not configured, the source is *null*. For the current release, the null source is presumed to be the internal oscillator.

Possible Sources

The *source* of the clock has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. More typically, the source has the two-part, short-hand form *slot.line* or *slot.port*. If the source is an AXSM, the format is *slot.line*.

For a BITS clock, the format is *slot.port*. The slot for a BITS clock is 7. The logical port is always 35 or 36. Port 35 refers to the upper external clock connector, and port 36 refers to the lower connector.

Clock Status

The status of a particular clock source can be one of the following:

- “ok” (good), which means the clock source is operational and stable.

  (If the *status* is “ok,” then the *Reason* field shows “okay.” If the status is “ok,” the reason for the status change described in the section, “Reason for Status Change,” is not important.)
- “bad” means a fault in the clock source has been detected. Use the Reason field to help isolate the problem. See the section, “Reason for Status Change.”
- “unknown” is a temporary string while the clock manager is sending a message to the clock source.
- “not configured” means that this source—primary or secondary—has not been configured.

Reason for Status Change

The reasons that clock status can change are numerous. The **dspclksrcs** command displays a Reason field for both the primary and the secondary clock source. The reason can include the first-time, user-specification of the clock source. The reason strings and their meaning appear in Table 9-3. Additional information about “okay” and the locking process follows.

Table 9-3 Reasons for Change of Clock State

Reason	Meaning
okay	The clock source is okay.
unknown reason	The clock manager has no information for Reason.
no clock signal	Loss of signal (LOS) on the clock source.
frequency too high	The frequency has drifted too high.
frequency too low	The frequency has drifted too low
excessive jitter	Jitter has exceeded tolerance for this stratum.
missing card or component	The active PXM45 has no clock hardware support.
non-existent logical interface	The interface is non-existent or not functioning.
interface does not support clocking	The interface does not support clocking.
phase error	The clock manager has detected a phase error in the clock.
unlockable	The clock manager has attempted to lock the source but found that the clock signal from this source is unlockable.
out of lock or null	The clock circuitry is again trying to lock a source that went out of locking range. Note: for Reason, out of lock and null are the same.
reset—not a valid state	The clock source has been reset.
in locking—wideband test	The clock circuitry is in wide bandwidth mode of the locking process. In this mode, the circuit tests the integrity of the source but with wide latitude for frequency accuracy. If the source passes this test, the circuit proceeds to the narrowband test.
in locking—narrow-band test	The clock circuitry is in narrow bandwidth locking mode. In this mode, the circuit stringently tests the integrity of the source.
locked	The clock circuitry is locked to this source.

When you configure a new clock source or the current clock source changes due to any reason, the software goes through the process of validating the new, *current* clock source again. (For example, the reasons other than direct user-configuration can be: the previous clock source goes out of lock or a resync of the clock sources takes place due to a switch-over or a rebuild.) This validation process takes the current clock source through the following states:

- in locking—wideband test
- in locking—narrowband test
- locked

During these states, the node is already using the new clock source as the synchronizing source.

You might also see these states—in the sequence previously listed—if the current clock source was momentarily lost because it drifted out of the lockable range for either the frequency or the phase. In such a case, the software goes through one more round of trying to confirm that the current clock source is lockable before it declares a clock source to be unlockable. If the software finds that, even after this repeated attempt, that the clock source is not coming back within the lockable range, it declares the clock source as unlockable and proceeds to use the next clock in the hierarchy (of primary, secondary, internal oscillator) as the current clock source. The exception to this final validation scenario occurs if the current

clock source is the internal oscillator in either the *free running* mode or the *hold-over* mode: in this case, the software omits this final validation attempt because no other clocks sources within a clock hierarchy are available.

Revertive Behavior

For information on revertive behavior, see the **cnfclksrc** description.

Cards on Which This Command Runs

PXM45

Syntax

dspclksrcs

Related Commands

cnfclksrc, **delclksrc**, **dspclkalms**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the clocks. The display shows that both the primary and secondary clocks are good. They are sourced at lines 2 and 3 of the AXSM in slot 6. Also, the primary source is providing the active clock, and the system is configured for revertive behavior. The primary and secondary clock reasons are okay.

```
pinnacle.7.PXM.a> dspclksrcs
Primary clock type:     generic
Primary clock source:   6.2
Primary clock status: good
Primary clock reason:   okay
Secondary clock type:   generic
Secondary clock source: 6.3
Secondary clock status:   good
Secondary clock reason:   okay
Active clock:           primary
source switchover mode: revertive
```

Display information about the clock sources. This example shows a BITS clock for the primary source with revertive mode enabled.

```
pop20one.7.PXM.a > dspclksrcs
Primary clock type:     bits t1
Primary clock source:   7.35
Primary clock status:   ok
Primary clock reason:   okay
Secondary clock type:   generic
Secondary clock source: 9:1.1:1
Secondary clock status:   ok
Secondary clock reason:   okay
Active clock:           primary
source switchover mode: revertive
```

dspcon

Display Connection

Display information about an SPVC. The contents of the display on the AXSM and the PXM45 differ slightly. On both cards, the **dspcon** output appears in sections to make the information easier to sort.

Most of the information in the **dspcon** output comes from **addcon** execution. See the **addcon** description for more information. Also, executing **cnfpnni-intf** can affect the **dspcon** output.

Display Connection on the PXM45

On the PXM45, **dspcon** shows the following connection identifiers:

- NSAP address, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type and compliance (for example, UBR for service type and UBR.1 for ATM Forum compliance).
- Bearer class (relates to voice traffic and is reserved for future use).
- Whether continuity checking or frame discard are enabled (see **addcon** description).
- Cause of the last failure. This field can also show that no errors have occurred since the connection was first added by displaying “SPVC Established.” If a failure occurred, the Attempts field shows the number of times the system attempted to re-establish service. If no failures have occurred, the Attempts field contains a 0.
- L-Util and R-Util are the local and remote percent of utilization assigned to the connection. Currently, the default of 100% is the only value.
- Cost values for the connection’s route: the two fields in this category are Max Cost and Routing Cost. The Max Cost is a cost-per-link configured for a service type (such as UBR) through the **cnfpnni-intf** command. When you add the SPVC through **addcon**, you can specify a maximum routing cost through the maximum cost (maxcost) parameter. The maxcost represents the maximum cost for an individual connection. The system uses the cost-per-link for the service type and the maxcost for the connection to determine whether a route costs too much. After the system creates a route, the total number of links yields the Routing Cost.

  The default cost-per-link is 5040, so if a particular service type uses the default and a route consists of 4 links, the Routing Cost is 20160. If the **dspspvc** display shows that Max Cost is –1, no limit was specified through **cnfpnni-intf**, and the resulting Routing Cost is 0.
- Broadcast type: point-to-point or multicast.

The Traffic Parameters section shows the standard parameters PCR, SCR, and CDV in the receive and transmit directions.

Display Connection on the AXSM

On the AXSM, **dspcon** shows the following connection identifiers:

- NSAP address, logical port, VPI/VCI, status, and ownership of local and remote ends of the connection. The display shows whether a particular endpoint is the master or slave.

The provisioning parameters in the display show:

- Connection type of VPC or VCC.
- Service type (for example, ABR).
- A number indicating the controller. For example, 2 refers to PNNI. The **addcontroller** command specifies the controller.
- The administrative state is either up or down. This state results from **addcon** or **dncon**/**upcon**. Note that, after you down a connection with at the connection master endpoint, the **dspcon** command shows the connection as “down” when you execute it at the master endpoint and “failed” when you execute it at the slave endpoint. (See also **dncon** description).
- The operational state is either OK or failed. The operational state can apply to a connection regardless of the administrative state.

The traffic management parameters consist of:

- Local and remote UPC parameters of PCR, MBS, CTD, CDVT, and so on. A –1 in a field means that the parameter was not specified. The characters “N/A” indicate that the parameter does not apply to the service type.

These other fields also pertain to connection integrity:

- OAM connectivity check enable or disable.
- Loopback test enable/disable and loopback type.
- Round trip delay in microseconds. This field is non-zero only if you previously executed **tstdelay**.

The **dspcon** command requires a unique connection identifier. If you do not have the information to identify a connection, execute **dspcons**. On the AXSM, **dspcons** identifies all the connections on the AXSM. On the PXM45, **dspcons** identifies all the connections on the node. (See **dspcons** description).

and Egress directions.

Cards on Which This Command Runs

AXSM, PXM45

AXSM Syntax

dspcon
<*ifNum*>
<*vpi*>
<*vci*>

PXM45 Syntax

dspcon
<*portid*>
<*vpi*>
<*vci*>

Syntax Description

ifNum	(AXSM) Logical interface (port) number. For AXSM, the range is 1–60.
portid	(PXM45) The *portid* represents the PNNI logical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.
vpi	VPI number. At the UNI, the range is 0–255. At the NNI, the range is 0–4095.
vci	VCI number. For a VCC, the range is 32–65535. For a VPC, the VCI is 0.

Related Commands

addcon, **dspcons**, **cnfcon**

PXM45 Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active	Privilege: GROUP1

Examples

Display connection 5 31 63000 on the current AXSM.

```
MGX8850.1.AXSM.a > dspcon 5 31 63000
--------------------------------------------------------------------------

Local   :               NSAP  Address               port     vpi    vci

(M)       47009181000000000107BE92F3F00000101180500  1.01.05 31 63000
Remote  :               NSAP  Address               port     vpi    vci

(S)       47009181000000000107BE92F3F00000101180500  1.01.05   3201 100
--------------------------------------------------------------------------

Conn. Type    :     VCC                          Admn Status  : ADMN-UP
Service Type  :    cbr1                          Oper Status  : FAIL
Controller    :       2
--------------------------------------------------------------------------

Local PCR     :   10000                          Remote PCR   : 1000
Local SCR     :     N/A                          Remote SCR   : N/A
Local CDV     :      -1                          Remote CDV   : -1
Local CTD     :      -1                          Remote CTD   : -1
Local MBS     :     N/A                          Remote MBS   : N/A
Local CDVT    :      -1                          Remote CDVT  : -1
Admin weight  :      -1                          Frame discard: N
--------------------------------------------------------------------------

OAM CC Config :DISABLED                          Statistics   : DISABLED
```

```
-------------------------------------------------------------------------

Loopback Type :No  Lpbk | Dir:N/A     | Status: No Lpbk | RTD: 0 us
```

On the CLI of the PXM45, display connection 20 100 on 11:1.1:2.

```
Unknown.7.PXM.a > dspcon 11:1.1:2 20 100
Port                   Vpi Vci                         Owner      State
-------------------------------------------------------------------------
Local  11:1.1:2        20.100                          MASTER     FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00
Remote 11:1.1:2        10.100                          SLAVE      FAIL
       Address: 47.00918100000000107b65f33d.0000010b1802.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: CBR         Conformance: CBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                 Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  50            Rx PCR:  50
Tx CDV:  N/A           Rx CDV:  N/A
Tx CTD:  N/A           Rx CTD:  N/A
```

Display information for vpi/vci 10 100 on port ID 1:1.1:1. In this case, port ID and remote and local NSAP addresses are the same, so the connection is a DAXCON. Also, the Max Cost is –1. The Max Cost of -1 means no cost-per-link was specified for UBR service, and therefore the Routing Cost is 0.

```
node19.8.PXM.a > dspcon 1:1.1:1 10 100
Port                   Vpi Vci                         Owner      State
-------------------------------------------------------------------------
Local  1:1.1:1         10.100                          SLAVE      OK
       Address: 47.009181000000000001a53c82d.000001011801.00
Remote 1:1.1:1         11.101                          MASTER     OK
       Address: 47.009181000000000001a53c82d.000001011801.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: UBR         Conformance: UBR.1
Bearer Class: BCOB-X
Last Fail Cause: SPVC Established                 Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0

---------- Traffic Parameters ----------
Tx PCR:  14            Rx PCR:  14
Tx SCR:  3             Rx SCR:  3
Tx MBS:  1             Rx MBS:  1
Tx CDVT: -1            Rx CDVT: -1
Tx CDV:  N/A           Rx CDV:  N/A
Tx CTD:  N/A           Rx CTD:  N/A
```

dspconinfo

Display Summaries of Connection Information—display general information about SPVCs.

The **dspconinfo** runs on the PXM45 and lists the total number SPVCs on each PNNI port on the node. the default is all ports on the node, but you can also specify a particular PNNI port. For each port, the display shows:

- Number of active connections
- Number of failed connections
- Number of administratively inactive (or down) connections as a result of **dncon**
- Number of alarm indication (AIS) failures
- Number of A-bit failures
- Total number of connections

Note that **dspconinfo** checks for the down state at the master end of the connection (status at the slave endpoint is failed). See also the descriptions for **dncon** and **dspcon**.

Cards on Which This Command Runs

PXM45

Syntax

```
dspconinfo
[-port portid]
```

Syntax Description

-port	Specifies a particular port for the connections. The default for this command is all ports (*portid* not specified) You can specify a particular port by using the **port** keyword followed by the portID in the following format: [*shelf.*]*slot*[:*subslot*].*port*[:*subport*] Currently, the value for shelf is 0 and so is not necessary.

Related Commands

none

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the SPVC summaries for the current node.

```
svcswp10.7.PXM.a > dspconinfo

Local Port     #Active      #Fail   #AISFail  #ABITFail      #Down     #Total
----------------------------+-----------------------------+---------------------
5.4               3998          0          0          0          0       3998
6.4               3997          0          0          0          0       3997
```

An example of an individual port follows. The display shows 1096 connections on 11:1.1:11.

```
Geneva.7.PXM.a > dspconinfo -port 11:1.1:11

Local Port      #Active   #Failed     #Total
---------------------------------------------
11:1.1:11        1096          0       1096
```

dspcons

Display Connections—display basic information for all connections.

The default usage of **dspcons** uses no parameters and causes all available information for the connections to appear. To narrow the scope of the output, use one or more optional parameters.

The **dspcons** command runs on the CLI of either the AXSM or the PXM45. The set of optional parameters and the output are different on these cards. (See Syntax Description for the card-specific parameters.) On the AXSM, the columns at the head of the information fields are:

record	A number for the connection with internal application only. It resides in the database on the AXSM and is not affected by user input. The system creates this number when you create the connection. The Cisco WAN Manager application uses this number.
Identifier	Identifies the connection in the format *port vpi vci*.
Type	Shows whether the connection is a VCC or a VPC.
SrvcType	The service type—VBR, and so on. (See **addcon** description).
M/S	Indicates whether the endpoint specified by *Identifier* is the master or slave.
Upld	The hexadecimal Upload number is an encoded timestamp the Cisco WAN Manager application uses to determine when a connection was created or modified. In the CLI context, this field has little meaning.
Adm	The administrative state of the connection. If the connection is down, it may have resulted from the **dncon** command.
Alarm	Shows the alarm status of the connection.

When you execute **dspcons** on the CLI of the PXM45, the output shows:

Local and Remote Port ID	The display contains a column for the local port ID and a column for the remote port ID. The port ID has the format that the network controller utilizes: [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]
Local and Remote VPI.VCI	The VPI and VCI at the local and remote ends of the connection.
State	The State column shows whether the connection is OK, down (by the **dncon** command), failed, or has an alarm indication signal (AIS) or *abit* errors.
Owner	Whether the endpoint is master or slave.
Local and Remote NSAP	An NSAP address for each end of the connection.

Cards on Which This Command Runs

PXM45, AXSM

PXM45 Syntax

```
dspcons
[-port portid]
[-vpi starting-vpi]
[-vci starting vci]
[-state {fail|ais|abit|ok|down}]
[-owner {master|slave}]
```

PXM45 Syntax Description

-**port**	The port identifier (*portid*) in the format that the network controller utilizes: [*shelf*.]*slot*[:*subslot*].*port*[:*subport*] Currently, the value for *shelf* is not necessary.
-**vpi**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**vci**	The VPI of the connection that you would like to serve as the starting connection in the display.
-**state**	A specific connection state. The display shows only the connections with the state you specify. Note that on the PXM45, you must spell out the entire state keyword. The keywords for specifying a state are **failed**—only failed connections **ais**—connections with alarm indication signal (AIS) set **abit**—connections on which an A-bit error has occurred **ok**—connections with no problems **down**—connections that are administratively down because a user has executed **dncon** to down the connection
-**owner**	Specifies connections where the local endpoint is either master or slave.

AXSM Syntax

```
dspcons
[-conn <conn id>]
[-filt <filter options>]
[-if <intf no>]
[-vpi <vpi filter>]
[-vci <vci filter>]
```

AXSM Syntax Description

-conn	The connection ID (*conn ID*) of the connection to begin the display. The format of *conn ID* is: *ifNum.vpi.vci* The range for *ifNum* is 1–60 for the AXSM. The VPI has the range 0–4095. The VCI has the range 32–65535.
-filt	Unlike on the PXM45, you do not use keywords for this parameter on the AXSM. You enter only a number on the AXSM CLI to indicate the state. The display criteria are: 1 ingr—for errors in the ingress direction 2 egr—for errors in the egress direction 3 condn—for connections where the switch has conditioned the connection 4 iffail—for connection on a failed logical interface 5 ccfail 6 mis 7 abit—for connections where an A-bit error has occurred
-if	A particular logical interface for connection display.
-vpi	The VPI of all the connections that you would like to display.
-vci	The VCI of all the connections that you would like to display.

Related Commands

dspcon, **addcon**, **cnfcon**, **delcon**, **dncon**, **upcon**

PXM45 Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

AXSM Attributes

Log: no log	State: active, standby	Privilege: GROUP1

PXM45 Example

Display all connections by entering **dspcons** on the CLI of the PXM45.

```
MGX8850.7.PXM.a > dspcons

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
3:1.1:1            20 0      6:1.1:1            20 0       OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001031801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
5:1.1:1           100 100    5:1.1:1           100 200     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
5:1.1:1           100 200    5:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001051801.00
Remote Addr: 47.00918100000000107b65f33d.000001051801.00
6:1.1:1            20 0      3:1.1:1            20 0       OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001031801.00
6:1.1:1           100 100    6:1.1:1           100 200     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           100 200    6:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK          SLAVE

6:1.1:1           100 200    6:1.1:1           100 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
6:1.1:1           200 100    6:2.1:3           200 200     OK          SLAVE

Local  Addr: 47.00918100000000107b65f33d.000001061801.00
Remote Addr: 47.00918100000000107b65f33d.000001061803.00
6:2.1:3           200 200    6:1.1:1           200 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.000001061803.00
Remote Addr: 47.00918100000000107b65f33d.000001061801.00
9:1.3:3            10 100    Routed              0 0       FAIL        SLAVE
Local  Addr: 47.00918100000000107b65f33d.000001091803.00
Remote Addr: 00.000000000000000000000000.000000000000.00
11:1.1:2           10 100    11:1.1:2           20 100     OK          SLAVE
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00

Local Port         Vpi.Vci   Remote Port        Vpi.Vci    State   Owner
----------------------------+----------------------------+-------+------
11:1.1:2           20 100    11:1.1:2           10 100     OK          MASTER
Local  Addr: 47.00918100000000107b65f33d.0000010b1802.00
Remote Addr: 47.00918100000000107b65f33d.0000010b1802.00
```

AXSM Example

Display all connections on the current AXSM. In this example, only one connection exists. Master and slave endpoints are shown.

```
GN.6.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld    Admn   Alarm
------    ----------   ----   --------   ---    ----    ----   -----
    0  01.0010.00100   VCC        cbr1   S   010c7953    UP      none
    1  04.0020.00100   VCC        cbr1   M   010c7964    UP      none
```

dspcons-dbg

Display Connections-Debug—shows whether the SPVC log is enabled or disabled.

The **dspcons-dbg** command shows whether the SPVC log is enabled. The **dbgcon** command lets you enable or disable this log.

Cards on Which This Command Runs

PXM45

Syntax

`dspcons-dbg`

Syntax Description

This command takes no parameters.

Related Commands

dbgcon

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Determine whether SPVC logging is enabled. The output shows that the log is disabled.

```
MGX8850.8.PXM.a > dspcons-dbg

Global SPVC Event Log Status: Disable
```

dspcontrollers

Display Controllers

Displays all controllers that have been added through the **addcontroller** command.

Cards on Which This Command Runs

PXM45

Syntax

dspcontrollers

Syntax Description

This command takes no parameters.

Related Commands

addcontroller, **delcontroller**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display all controller. In this example, the switch has only one controller—PNNI. The display also shows that the controller is internal (slot 7) and has the optional, user-specified name PNNITWO. Apart from controller information, the display shows that no shelf alarms exist.

```
pop20two.7.PXM.a > dspcontrollers
pop20two                          System Rev: 02.00   Jul. 30, 2000 09:39:36 GMT
MGX8850                                               Shelf Alarm: NONE
Number of Controllers:        1
Controller Name:              PNNITWO
Controller Id:                2
Controller Location:          Internal
Controller Type:              PNNI
Controller Logical Slot:      7
Controller Bay Number:        0
Controller Line Number:       0
Controller VPI:               0
Controller VCI:               0
Controller In Alarm:          NO
Controller Error:
```

dspdiagcnf

Display Diagnostics Configuration

Displays the current diagnostics configuration, such as whether online or offline is enabled, the coverage time, starting time, and the days of the week (SMTWTFS) that the offline diagnostics runs.

Coverage indicates the length of time that the diagnostics will run as follows:

- light = 5 minutes or less
- medium = 30 minutes or less
- full = 60 minutes or more

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

```
dspdiagcnf
```

Syntax Description

This command takes no parameters.

Cards on Which This Command Runs

PXM45

Related Commands

cnfdiag, cnfdiagall

Attributes

Log: no log	State: active, standby	Privilege: SERVICE_GP

Example

```
pop20one.7.PXM.a > dspdiagcnf
Online   -------------- Offline -------------
SlotEnableEnableCoverageStartTimeSMTWTFS

1enableenablefull10:30-M-W-F-
2disableenablefull22:00-M-W-F-
3enabledisablemedium11:15-M-W-F-
4enabledisablelight03:20-M-W-F-
5enabledisablelight14:30------S
6disableenablefull12:00-M-W-F-
7disableenablelight18:00------S
8enableenablemedium05:00-M-W-F-
9enabledisablelight04:30-M-W-F-
10enabledisablefull22:00------S
11enabledisablelight24:15-M-W-F-
12disableenablefull11:00------S
13enableenablelight13:15-M-W-F-
14enableenablemedium01:00-M-W-F-
15enableenablefull02:30-M-W-F-
16disabledisablelight00:00SMTWTFS
17disabledisablelight00:00SMTWTFS
18disabledisablelight00:00SMTWTFS
19disabledisablelight00:00SMTWTFS
20disabledisablelight00:00SMTWTFS
21disabledisablelight00:00SMTWTFS
22disabledisablelight00:00SMTWTFS
23disabledisablelight00:00SMTWTFS
24disabledisablelight00:00SMTWTFS
25disabledisablelight00:00SMTWTFS
26disabledisablelight00:00SMTWTFS
27disabledisablelight00:00SMTWTFS
28disabledisablelight00:00SMTWTFS
29disabledisablelight00:00SMTWTFS
30disabledisablelight00:00SMTWTFS
31disabledisablelight00:00SMTWTFS
32disabledisablelight00:00SMTWTFS
```

dspdiagerr

Display Diagnostics Errors

Displays the current offline or online diagnostics errors.

Cards on Which This Command Runs

PXM45

Syntax

dspdiagerr <*online/offline*>

Syntax Description

online/offline	Specify whether to display the online or the offline diagnostics errors.

Related Commands

clrdiagerr

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

```
MGX8850.7.PXM.a > dspdiagerr offline
Slot Date   Time  Message
---- ----   ----  -------
1    --     --
2    24-Aug 13:35 SAR send failed
3    --     --
4    --     --
5    --     --
6    --     --
7    --     --
8    --     --
9    --     --
10   --     --
11   --     --
12   --     --
13   18-Aug 03:24 HUMVEE IRQ test failed
14   --     --
15   --     --
16   --     --
17   --     --
18   --     --
19   --     --
20   --     --
21   --     --
22   --     --
23   --     --
```

```
24   --     --
25   --     --
26   --     --
27   --     --
28   --     --
29   --     --
30   --     --
31   --     --
32   --     --
```

dspdiagstat

Display Diagnostics Statistics

Displays the number of times that the diagnostics has run. The output shows the number of attempts and the number of failures for both offline and online diagnostics.

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

dspdiagstat
<*slot*>

Syntax Description

slot	The slot of the card for which to display the diagnostics statistics.

Cards on Which This Command Runs

PXM45

Related Commands

clrdiagstat

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

```
pop20two.7.PXM.a > dspdiagstat 7

Slot 7 diagnostics statistics:

online diag attempted  = 0x00000000
online diag failed     = 0x00000000
offline diag attempted = 0x00000000
offline diag failed    = 0x00000000
```

dspdiagstatus

Display Diagnostics Status

Displays the diagnostics status for each card on the switch. The diagnostics statuses are:

- Idle
- Ready
- Offline
- Online

Note See the **cnfdiag** command for a detailed description of MGX 8850 diagnostics.

Syntax

`dspdiagstatus`

Syntax Description

This command takes no parameters.

Cards on Which This Command Runs

PXM45

Related Commands

cnfdiag, cnfdiagall

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

```
pop20one.7.PXM.a > dspdiagstatus
SlotStateRole

1ReadyACTIVE CARD ROLE
2ReadyACTIVE CARD ROLE
3OfflineSTANDBY CARD ROLE
4OnlineACTIVE CARD ROLE
5ReadyACTIVE CARD ROLE
6OfflineSTANDBY CARD ROLE
7ReadyACTIVE CARD ROLE
8OnlineACTIVE CARD ROLE
9IdleACTIVE CARD ROLE
10ReadyACTIVE CARD ROLE
11OnlineACTIVE CARD ROLE
12IdleACTIVE CARD ROLE
13ReadyACTIVE CARD ROLE
14OfflineSTANDBY CARD ROLE
15IdleACTIVE CARD ROLE
```

```
16IdleACTIVE CARD ROLE
17IdleACTIVE CARD ROLE
18IdleACTIVE CARD ROLE
19IdleACTIVE CARD ROLE
20IdleACTIVE CARD ROLE
21IdleACTIVE CARD ROLE
22IdleACTIVE CARD ROLE
23IdleACTIVE CARD ROLE
24IdleACTIVE CARD ROLE
25IdleACTIVE CARD ROLE
26IdleACTIVE CARD ROLE
27IdleACTIVE CARD ROLE
28IdleACTIVE CARD ROLE
29IdleACTIVE CARD ROLE
30IdleACTIVE CARD ROLE
31IdleACTIVE CARD ROLE
32IdleACTIVE CARD ROLE
```

dspenvalms

Display Environment Alarms

Display alarms related to the environment of the node. The monitored categories are:

- Temperature inside the enclosure
- AC power supplies if applicable
- DC supply power
- DC system power
- Bottom fan tray operation
- Top fan tray operation

The definition of each alarm severity comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a line being disconnected is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

The **dspenvalms** command is part of a hierarchy of troubleshooting commands you can execute on the PXM45. Frequently, **dspenvalms** follows the higher-level command **dspndalms**. The **dspndalms** command shows a variety of alarm types within the switch and helps isolate the problem.

Cards on Which This Command Runs

PXM45

Syntax

dspenvalms
[*temp*]
[*psu*]
[*fan*]
[*vmon*]

Syntax Description

temp	(temperature) shows the temperature and whether an alarm condition exists.
psw	(power supply units) shows how many AC power supplies reside in the power supply tray, and also shows the high and low DC output values that the AC power supplies should be able to maintain.
fan	shows the presence of top and bottom fan trays, minimum rotation rate of each fan, and actual rotation rate of each fan in RPMs.

vmon (voltage monitor) shows permitted ranges and actual DC voltages.

Related Commands

dspndalms, **dspcdstatus**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Check the temperature.

```
MGX8850.7.PXM.a > dspenvalms temp
MGX8850                          System Rev: 02.00   Aug. 06, 2000 18:28:33 GMT
MGX8850                                              Node Alarm: CRITICAL
ENVIRONMENTAL ALARM STATE INFO   ^Notification Disabled
   Alarm Type      Unit   Threshold      DataType   Value       State
 ----------------  ---- --------------   -------- ---------- -------------
 Temperature             <= 50            Celsius   33       Normal
```

Check the fans for alarms. The display shows minimal and actual RPMs for each fan. The display shows that the bottom fan tray is missing.

```
MGX8850.7.PXM.a > dspenvalms fan
MGX8850                          System Rev: 02.00   Aug. 06, 2000 22:33:08 GMT
MGX8850                                              Node Alarm: CRITICAL
ENVIRONMENTAL ALARM STATE INFO   ^Notification Disabled
   Alarm Type      Unit   Threshold      DataType   Value       State
 ----------------  ---- --------------   -------- ---------- -------------
 Top Fan Tray       1    >= 2000          RPM       3504     Normal
 Top Fan Tray       2    >= 2000          RPM       3498     Normal
 Top Fan Tray       3    >= 2000          RPM       3576     Normal
 Top Fan Tray       4    >= 2000          RPM       3492     Normal
 Top Fan Tray       5    >= 2000          RPM       3474     Normal
 Top Fan Tray       6    >= 2000          RPM       3564     Normal
 Top Fan Tray       7    >= 2000          RPM       3462     Normal
 Top Fan Tray       8    >= 2000          RPM       3366     Normal
 Top Fan Tray       9    >= 2000          RPM       3444     Normal

 Bottom Fan Tray    1    >= 2000          RPM       0        Missing
 Bottom Fan Tray    2    >= 2000          RPM       0        Missing
 Bottom Fan Tray    3    >= 2000          RPM       0        Missing
 Bottom Fan Tray    4    >= 2000          RPM       0        Missing
 Bottom Fan Tray    5    >= 2000          RPM       0        Missing
 Bottom Fan Tray    6    >= 2000          RPM       0        Missing
 Bottom Fan Tray    7    >= 2000          RPM       0        Missing
```

Display all environment alarms for the enclosure by entering **dspenvalms** with no optional parameters.

```
MGX8850.7.PXM.a > dspenvalms
MGX8850                          System Rev: 02.00   Aug. 06, 2000 18:20:35 GMT
MGX8850                                              Node Alarm: CRITICAL
ENVIRONMENTAL ALARM STATE INFO   ^Notification Disabled
   Alarm Type      Unit   Threshold      DataType   Value       State
 ----------------  ---- --------------   -------- ---------- -------------
 Temperature             <= 50            Celsius   33       Normal
```

```
 Power Supply       A1   none             None      none     Normal
 Power Supply       A2   none             None      none     Normal
 Power Supply       A3   none             None      none     Normal
 DC Voltage         A    42  to 54        VoltsDC   49       Normal

 Power Supply       B1   none             None      none     Missing
 Power Supply       B2   none             None      none     Missing
 Power Supply       B3   none             None      none     Missing
 DC Voltage         B    42  to 54        VoltsDC   0        Normal

 Top Fan Tray       1    >= 2000          RPM       3504     Normal
 Top Fan Tray       2    >= 2000          RPM       3498     Normal
 Top Fan Tray       3    >= 2000          RPM       3576     Normal
 Top Fan Tray       4    >= 2000          RPM       3498     Normal
 Top Fan Tray       5    >= 2000          RPM       3480     Normal
Top Fan Tray       6    >= 2000          RPM       3570     Normal
 Top Fan Tray       7    >= 2000          RPM       3468     Normal
 Top Fan Tray       8    >= 2000          RPM       3366     Normal
 Top Fan Tray       9    >= 2000          RPM       3444     Normal

 Bottom Fan Tray    1    >= 2000          RPM       0        Missing
 Bottom Fan Tray    2    >= 2000          RPM       0        Missing
 Bottom Fan Tray    3    >= 2000          RPM       0        Missing
 Bottom Fan Tray    4    >= 2000          RPM       0        Missing
 Bottom Fan Tray    5    >= 2000          RPM       0        Missing
 Bottom Fan Tray    6    >= 2000          RPM       0        Missing
 Bottom Fan Tray    7    >= 2000          RPM       0        Missing
 Bottom Fan Tray    8    >= 2000          RPM       0        Missing
 Bottom Fan Tray    9    >= 2000          RPM       0        Missing

 +5V Input             4.850^ to 5.150^   VoltsDC   5.017    Informational
 +3.3V Input           3.200^ to 3.400^   VoltsDC   3.259    Informational
+2.5V Input           2.425^ to 2.575^   VoltsDC   2.440    Informational
 Calibration VDC        0x7e^ to 0x82^    Other     0x80     Informational
```

dsperr

Display Error

Display error message files. The **dsperr** command is primarily a debug command. Because it displays tasks and system calls, the information is more suitable to developers or others who can use information that is internal to the switch rather than applicable to the network. The information may also be useful to Cisco support personnel.

Cards on Which This Command Runs

PXM45

Syntax

`dsperr <-s1 slot> [-en <Error#>] [ -tr {P|L|N}]`

Syntax Description

-sl (Mandatory) Number of the slot. The value of *slot* is any slot in the switch.

-en An option that lets you specify a particular error record. You can list all the errors by using **dsperr** without this parameter and note the Error Num then display just that error record.

-tr This option allows you specify 3 options for printing trace data: P, L, or N. If **-tr** is not specified, the trace data is printed normally.

P–Pause prompts before printing the trace data with the following message:

This section contains trace data that may span multiple pages. This data is contained in the file:

C:\LOG\slot09\error08.log

You can ftp this file to a workstation for further analysis

Do you want to view this data now [Yes/No]?

L–Lists all of the trace data file names, for example:

C:\LOG\slot09\error08.log

C:\LOG\slot09\error07.log

C:\LOG\slot09\error06.log

C:\LOG\slot09\error09.log

N–No disables trace data printing.

Related Commands

clrerr

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display one screen of internal error messages for slot 1. The Error Num is 4849.

```
pop20two.8.PXM.a > dsperr -sl 1

Error Log for Slot 01: Error Num 4849
    Firmware version:                          Product Id: 8850
    Timestamp: 02/20/2001-14:04:42 Node name: pop20two
Section Number 0:
Event Logged:
 01-00266 02/20/2001-14:04:38 SHMA-7-API_BC_REPORT
 E:04849 tEmFaultMg 0x801d2fc0
 shmBackCardInsertReport: AppId 0x10005, tId 0x1003d, tName tEmFaultMgr , BayTyp
e 1, callerPc 0x80237ff0


Section Number 1:
Stack Trace:
0x80694668 vxTaskEntry              +00c: sysTaskSetup+0()
0x80159068 sysTaskSetup             +0a4: emFaultMgrTaskEntry+0()
0x80237950 emFaultMgrTaskEntry      +114: ssiIpcComEpWait+0()
0x8016313c ssiIpcComEpWait          +070: ssi_ipc_mhdlr_receive_loop+0()
0x80165e34 ssi_ipc_mhdlr_receive_loop+100: ssi_ipc_mhdlr_msg_receive+0()
0x801666a8 ssi_ipc_mhdlr_msg_receive+054: fmFaultMgrTaskHandler+0()
0x80237434 fmFaultMgrTaskHandler    +0b0: cemaFmPhyProc+0()
0x8023bb68 cemaFmPhyProc            +068: emHandleHwStatusRpt+0()
0x80238e28 emHandleHwStatusRpt      +460: cemaFmReportBCInsert+0()
0x80237ff0 cemaFmReportBCInsert     +440: shmBackCardInsertReport+0()
0x801d2fc0 shmBackCardInsertReport  +07c: ssiEvent+0()
0x80142454 ssiEvent                 +284: ssiEventMsgReport+0()
0x80142840 ssiEventMsgReport        +0e4: ssiBramMsgReport+0()
0x80142be0 ssiBramMsgReport         +0f8: ssiStackTrace+0()

--------------
```

dsperrhist

Display Error History—displays a list of errors for a card slot.

The display consists of the following for each record in the history file:

- A number for the entry in the error history file
- An event number in hexadecimal format
- An event name—a few words that describe the error (such as the severity or affected area)
- A time stamp

The maximum number of entries in the history for a slot is 10. When the 11th error is logged, the switch deletes the oldest entry. Alternatively, you can clear the error history by executing **clrerrhist**.

If no entries exist in the history, the system returns the message, Nothing is logged.

Cards on Which This Command Runs

PXM45

Syntax

dsperrhist [*slot*]

Syntax Description

slot	Number of the slot—any slot in the switch. If you do not enter a slot number, the system displays the error history on the active PXM45.

Related Commands

clrerrhist

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Examples

Display the error history for the current PXM45. This PXM45 resides in slot 8.

```
jdlenoir.8.PXM.a > dsperrhist

Log of Errors and Failures (Slot 8):

Entry#--Event#---EventName-------------------------------TimeStamp-----------
General:
01      0x5006   CBC(Slave) Non-Fatal Minor              11/21/2000-17:55:49
02      0x5006   CBC(Slave) Non-Fatal Minor              11/21/2000-17:57:36
03      0x5006   CBC(Slave) Non-Fatal Minor              11/21/2000-18:22:47
-----------------------------------------------------------------------------
```

Display the error history for the AXSM in slot 6.

```
jbscott.8.PXM.a > dsperrhist 6

Log of Errors and Failures (Slot 6):

Entry#--Event#---EventName-------------------------------TimeStamp-----------
General:
01      0x0005   SHM_CDF_DISCOVER_TMR_POP                11/21/2000-13:49:47
-----------------------------------------------------------------------------
```

Display the error history for the AXSM in slot 1.

```
pop20two.7.PXM.a > dsperrhist 1

Log of Errors and Failures (Slot 1):

Entry#--Event#---EventName-------------------------------TimeStamp-----------
General:
01      0x0005   SHM_CDF_DISCOVER_TMR_POP                11/29/2000-16:45:18
02      0x0001   SHM_CDF_INVALID_SW_ID                   11/29/2000-16:45:48
03      0x0002   SHM_CDF_SW_DNLD_FAILED                  11/29/2000-16:51:17
04      0x0005   SHM_CDF_DISCOVER_TMR_POP                12/26/2000-16:23:19
Hardware Alarm:
01      0x5000   Humvee Non-Fatal Minor                  12/18/2000-16:48:36
02      0x5000   Humvee Non-Fatal Minor                  12/18/2000-16:49:11
03      0x5000   Humvee Non-Fatal Minor                  12/18/2000-16:49:38
04      0x5000   Humvee Non-Fatal Minor                  12/18/2000-16:54:31
05      0x5000   Humvee Non-Fatal Minor                  12/26/2000-16:16:34
-----------------------------------------------------------------------------
```

dspilmicnt

Display ILMI Counters

Displays the ILMI counters for a particular resource partition on a particular logical port.

Cards on Which This Command Runs

AXSM

Syntax

dspilmicnt <*ifNum*> <*partId*>

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
partId	The number of the resource partition. The range is 1–20.

Related Commands

cnfilmi, **dspilmi**, **dspilmis**, **clrilmicnt**, **dnilmi**, **upilmi**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the ILMI counters for logical port 1 on the current AXSM card.

```
pinnacle3.1.2.AXSM.a > dspilmicnt 1

  Port Num:                 1
  SNMPPDUsReceived:         0
  Get RequestsReceived:     0
  GetNextRequestsReceived:  0
  SetRequestsReceived:      0
  TrapReceived:             0
  GetResponseReceived       0
  GetResponseTransmitted:   0
  GetRequestTransmitted:    0
  TrapsTransmitted:         0
  InvalidPDUReceived:       0
  Asn1ParseError:           0
  NoSuchNameError:          0
  TooBigError:              0
```

dspln

Display Line

Display the configuration of a physical line.

Note The connection count includes control VCs when you execute **dspln** on the CLI of a service module. However, when you execute **dspcd** or **dsppnport**(**s**) on the CLI of the controller card, the display does not include control VCs.

Cards on Which This Command Runs

AXSM

Syntax

```
dspln
<-ds3 | -sonet> <bay.line>
```

Syntax Description

-ds3	Command delineator that precedes the *line number* entry for a T3 of E3 line.
-sonet	Command delineator that precedes the *line number* entry for a SONET line.
bay.line	Identifies the bay (1 or 2) and the number of the line. The line number can be 1 to the highest numbered line on the back card.

Related Commands

upln, **cnfln**, **delln**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display T3 line 1 on the current AXSM.

```
pinnacle.2.AXSM.a > dspln -ds3 1.2
  LineNum:               1.2
  LineEnable:            Down
  LineType:              ds3cbitadm
  LineCoding:            ds3B3ZS
  LineLength(meters):    0
  LineOOFCriteria:       fBits3Of8
  LineAIScBitsCheck:     Check
  LineLoopback:          NoLoop
  Xmt. Clock source:     localTiming
  LineRcvFEACValidation: 4 out of 5 FEAC codes
```

Display OC-48 line on the current OC-12 AXSM.

```
pop20two.1.AXSM.a > dspln -sonet 2.1
  Line Number            : 2.1
  Admin Status           : Up                Alarm Status        : Clear
  Loopback               : NoLoop            APS enabled         : Disable
  Frame Scrambling       : Enable            Number of ports     : 1
  Xmt Clock source       : localTiming       Number of partitions: 1
  Line Type              : sonetSts12c       Number of SPVC      : 0
  Medium Type(SONET/SDH) : SONET             Number of SVC       : 4
  Medium Time Elapsed    : 506223
  Medium Valid Intervals : 96
  Medium Line Type       : ShortSMF
```

dsplncnt

Display Line Counters—display the contents of counters for various cell-related statistics on a line.

The **dsplncnt** command displays counters for the following:

- CLP0 and CLP1 cells*
- Valid and errored OAM cells
- Invalid VPI/VCI/PTI cells*
- Last unknown VPI/VCI*
- Non-zero generic flow control (GFC) cells*
- Discarded or corrected HEC errors
- Discarded usage parameter control (UPC) cells with CLP0*
- Total UPC cells*
- Total non-compliant UPC cells*

An asterisk (*) indicates the displayed field does not apply to the AXSM-1-2488.

Note Some counters apply to both the ingress direction and the egress direction. Some apply to only the ingress direction. See the example output.

Cards on Which This Command Runs

AXSM

AXSM Syntax

dsplncnt *<bay.line>*

Syntax Description

<intvl>	The time interval to display (0–96). 0 is the current 15-minute interval. 1 is the most recent 15-minute interval. 2 is the next most recent 15-minute interval, and so on. 96 being the oldest 15-minute interval.
<bay>	Bay number: 1 or 2
<line>	Line number: • For OC12: 1 • For OC3: 1–4 • T3, E3: 1–8

Related Commands

clrlncnt

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the cell counters for line 1 in bay 1.

```
MGX8850.11.AXSM.a > dsplncnt 1.1
Line Number              : 1.1
                            Ingress       Egress
 CLP0 Cells               : 0                  0
 CLP1 Cells*              : 0                  0
 Valid OAM Cells*         : 0                  0
 Err OAM Cells*           : 0                  0
 Invalid VPI/VCI/PTI Cells*: 0                 0
 Non-zero GFC Cells*      : 0
 Last Unknown VPI*        : 0
 Last Unknown VCI*        : 0
 Discard HecErr Cells     : 115
 Corrected HecErr Cells   : 0
 NOTE: Counters with '*' do NOT apply to AXSM-1-2488 (OC48)
```

dsplns

Display Lines

Displays the configuration for all lines on a card. For each line, the output information consists of the:

- Bay and line number
- Line state—up (active) or down (inactive)
- The line type
- Whether any loopback currently exists on the line
- Line coding
- Frame scrambling status (enabled or disabled)
- Configured line length in meters (applies to only T3 or E3)
- Criteria for Out of Frame (OOF) error (applies to only T3 or E3)
- Whether C-bit (AIS) checking is enabled (applies to only T3 or E3)
- The medium line type—long reach, single-mode fiber, for example
- The alarm status—clear, critical, and so on

For information on an individual line, use **dspln**. Also, the **dspln** command shows the transmit clock configuration if one exists.

Cards on Which This Command Runs

AXSM

Syntax

dsplns

Related Commands

cnfln, **delln**, **dspcds**, **dspln**, **upln**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the configuration of the lines on an AXSM-4-622.

```
pop20two.1.AXSM.a > dsplns
                                          Medium Medium
  Sonet  Line     Line     Line    Frame   Line   Line    Alarm
  Line  State     Type     Lpbk   Scramble Coding Type    State
  ----- ----- ------------ ------ -------- ------ ------- ------
   1.1   Down sonetSts12c NoLoop   Enable  Other    Other Clear
   1.2   Down sonetSts12c NoLoop   Enable  Other    Other Clear
   2.1     Up sonetSts12c NoLoop   Enable  Other ShortSMF Clear
   2.2     Up sonetSts12c NoLoop   Enable  Other ShortSMF Critical
```

Examples

Display line configuration on the current AXSM-1-2488.

```
pinnacle.1.AXSM.a > dsplns
                                             Medium Medium
  Sonet  Line     Line     Line    Frame     Line   Line
  Line  Status    Type     Lpbk   Scramble   Coding Type
  ----- ----- ------------ ------  -------- -------- -------
   1.1   Down sonetSts48c NoLoop   Enable  Other ShortSingleMode
```

Display the configuration of each T3 line on the current AXSM-16-T3E3.

```
jdlenoir.11.AXSM.a > dsplns
  Line  Line      Line        Line     Length  OOF      AIS        Alarm
  Num   State     Type        Lpbk    (meters) Criteria cBitsCheck State
  ---- ------- ----------- ----------- ------  -------- ---------- -----
   1.1  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.2  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.3  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.4  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.5  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.6  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.7  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   1.8  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.1  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.2  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.3  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.4  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.5  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.6  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.7  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
   2.8  Down  ds3cbitadm      NoLoop        0 3Of8Bits      Check Clear
```

dsplog

Display Log

Display log file contents. The **dsplog** command is a debugging command and requires in-depth knowledge of the internal workings of the system. For example, the display may show points where the switch software steps into and out of functions or tracks tasks that it is spawning.

The PXM45 supports multiple log files: if the space allocation for one file becomes exhausted, the system starts filling a new, sequentially numbered file. The log files contain a substantial variety of information. With no parameters, the output contains all the contents of all the log files, so **dsplog** provides optional parameters for filtering the output. For example, you could specify only severe errors and only for a particular slot. The Syntax Description describes each parameter. Briefly, their functions are to:

- Specify an individual log file.
- Target a specific task.
- Specify an area of system functionality (called a *module* in the syntax).
- Display messages of a particular severity.
- Specify relative time periods in which errors may have occurred.

Cards on Which This Command Runs

PXM45

Syntax

```
dsplog
[-log <filename>]
[-sl <slot #>]
[-task <task name>]
[-mod <module name>]
[-sev <severity>]
[-tle <time same or earlier than>]
[-tge <time same or greater than>]
```

Syntax Description

-log	Specifies the *filename* of the error log. If you do not specify a file, the output scrolls through all log files one file at a time. To see a list of the existing log files, execute **dsplogs**.
-sl	Specifies the slot number for which to display errors.
-task	Specifies the task for which to display errors.
-mod	Specifies the module or functional area of switch software. The categories are numerous. Examples are: node alarm manager, card alarm manager, inter-process communications.

-sev	Specifies the severity of the alarm. Select it by entering a number in the range 1–7: 1. EVENT_FATAL: This severity indicates that the event affects the existing data traffic for the systems and is considered fatal because the platform cannot recover. Fatal events cause a card reset. Also, any error or condition that damages or causes loss of ongoing data traffic is fatal. Examples of fatal events are hardware watchdog timeout, critical task failure or suspension, and hardware device failures of CBC or QE. 2. EVENT_MAJOR_ALERT: This severity indicates a major service or feature of the platform has been damaged or lost but that existing data traffic is not affected. These events indicate that immediate action is necessary to recover the platform or service by posting traps and major alarms. Examples of major alerts are hard disk crashes, critical memory shortages, and inability to complete a configuration change. 3. EVENT_MINOR_ALERT: This severity indicates a minor event or partial damage to or loss of a service on the platform. Nevertheless, existing data traffic and critical services are not affected. These events indicate that eventual action is necessary to recover the platform or service by posting a minor alarm condition. Examples of minor alerts are loss of some tftp or telnet sessions and loss of statistics or other non-critical features. 4. EVENT_ERROR: This severity indicates that an error occurred but is not sever enough or it does not know the scope of its implication to be more sever. Most detected failures are reported with this severity, then the higher levels of software determine the appropriate response. Examples of these errors are *malloc* failures, illegal API parameter values, bad PDUs, and most internally detected failures. 5. EVENT_WARNING: This severity indicates that some threshold has been reached and could be a warning of a future error condition. Examples are resource shortages of memory and disk space, voltage and temperature just out of tolerance, and other conditions that could lead to a more serious situation. 6. EVENT_NOTICE: This severity indicates that a normal but significant event has occurred on the platform. Events for significant configuration changes would be in this category. Examples of notice type events would be addition of lines or ports and connection alarms. 7. EVENT_INFO: This severity indicates an event is informational. It does not indicate an abnormal condition. Examples of informational events are logging of user logins and important commands.
-tle	Specifies a particular time for which to display events: same time or earlier time.
-tge	Specifies a particular time for which to display events: same time or later time.

Related Commands

clrlog, **dsplogs**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the contents of log file number 1. This example shows the first of multiple screens.

```
jdlenoir.7.PXM.a > dsplog -log 1
07-00236 11/16/2000-15:21:26  CLI-7-CMDLOG
         tTnInTsk01 0x8037558c
 cliCmdLog: cisco@telnet.01: (cc 7).
11-00017 11/16/2000-15:21:26  CLI-7-CMDLOG
         tSmInTsk03 0x801ef16c
 cliCmdLog: cisco@smterm.03: (logout). - 1 dropped
07-00235 11/16/2000-15:18:26  CLI-7-CMDLOG
         tTnInTsk01 0x8037558c
 cliCmdLog: cisco@telnet.01: (cc 11).
11-00016 11/16/2000-15:18:25  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: cisco@smterm.02: (logout).
11-00015 11/16/2000-15:18:24  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: cisco@smterm.02: (cc 11).
07-00234 11/16/2000-15:18:18  CLI-7-CMDLOG
         tTnInTsk01 0x8037558c
 cliCmdLog: cisco@telnet.01: (cc 7).
07-00233 11/16/2000-15:18:17  CLI-7-CMDLOG
         tTnInTsk01 0x8037558c
 cliCmdLog: cisco@telnet.01: (login).
07-00232 11/16/2000-15:18:12  CLI-7-CLITNLOG
```

Display all logged events for slot 11. Note that the number 11 appears at the beginning of each log entry. This example shows the first of multiple screens.

```
jbscott.7.PXM.a > dsplog -sl 11
11-00032 11/16/2000-16:35:34  CLI-7-CMDLOG
         tSmInTsk03 0x801ef16c
 cliCmdLog: superuser@smterm.03: (logout). - 1 dropped
11-00031 11/16/2000-16:34:44  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: superuser@smterm.02: (logout).
11-00030 11/16/2000-16:34:44  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: superuser@smterm.02: (cc 11).
11-00029 11/16/2000-16:33:59  CLI-7-CMDLOG
         tSmInTsk03 0x801ef16c
 cliCmdLog: superuser@smterm.03: (logout). - 1 dropped
11-00028 11/16/2000-16:32:01  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: superuser@smterm.02: (logout).
11-00027 11/16/2000-16:32:00  CLI-7-CMDLOG
         tSmInTsk02 0x801ef16c
 cliCmdLog: superuser@smterm.02: (cc 11).
11-00026 11/16/2000-16:15:10  CLI-7-CMDLOG
         tSmInTsk03 0x801ef16c
 cliCmdLog: superuser@smterm.03: (logout). - 1 dropped
11-00025 11/16/2000-16:12:46  CLI-7-CMDLOG
```

dsplogs

Display Logs—display a list of all existing log files.

The **dsplogs** shows the existing log files (but not the file contents). The **dsplogs** command lets you see which files exist and thus determine specific contents to view when you execute **dsplog**.

Cards on Which This Command Runs

PXM45

Syntax

dsplogs

Syntax Description

This command takes no parameters.

Related Commands

dsplog

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

The output shows that one log file exists. To view the *contents* of the log file 1, you would enter **dsplog** -log 1. See description of **dsplog**.

```
jdlenoir.7.PXM.a > dsplogs

Current Event log number: 01
Event log 01:  C:/LOG/event01.log
   starting timestamp: 11/16/2000-13:29:13
```

dspndalms

Display Node Alarms

Displays various types of alarms on the node from a high-level perspective. With the information in the **dspndalms** display, you can select one of the following commands to investigate the alarm:

- **dspcdalms** identifies line, port, or connection alarms on an AXSM.
- **dspclkalms** shows alarms related to network clocks.
- **dspenvalms** lists alarms for out-of-range conditions for temperature, voltage sources, and so on.
- **dspslotalms** shows card-level alarms, such as a missing card or a disk problem on the PXM-HD.
- **dspswalms** shows alarms related to the switching fabric on the PXM45.

The preceding commands execute on the PXM45. If the results of the preceding commands seem to warrant it, you can **cc** to another card and execute alarm-specific or other troubleshooting commands. Use the **help** (or **?**) on the CLI of the other card to see available alarm commands (**? alm**, for example).

The definition of each *alarm severity* comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a line being disconnected is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

Cards on Which This Command Runs

PXM45

Syntax

```
dspndalms [slot]
```

Syntax Description

slot	The slot number of the card.

Related Commands

dspalm, **dspalms**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Display node alarms. The output shows two major, card-level alarms. Use the **dspcdalms** command to determine which card or cards have the alarms.

```
ITNODE3.7.PXM.a > dspndalms
Node Alarm Summary

Alarm Type                     Critical       Major         Minor
----------                     --------       -------       -------
Clock Alarms                         0             0             0
Switching Alarms                     0             0             0
Environment Alarms                   0             0             0
Card Alarms                          0             2             0
```

The output of **dspcdalms** shows that both card alarms are on the card in slot 8—the standby PXM45. the last line of the output recommends using **dspcdalms** with a specific slot to see more details.

```
ITNODE3.7.PXM.a > dspcdalms
Card Alarm Summary

Slot  Critical   Major     Minor
----  --------  -------   -------
 1         0       0        0
 2         0       0        0
 3         0       0        0
 4         0       0        0
 5         0       0        0
 6         0       0        0
 7         0       0        0
 8         0       2        0
 9         0       0        0
10         0       0        0
11         0       0        0
12         0       0        0
13         0       0        0
14         0       0        0

Use dspcdalms <slot> to see more detail.
```

Executing **dspcdalms** for slot 8 shows that a disk alarm and card state alarm exist.

```
ITNODE3.7.PXM.a > dspcdalms 8
Card Alarm Summary


Alarm Type                     Critical       Major         Minor
----------                     --------       -------       -------
Hardware Alarm                       0             0             0
Card State Alarm                     0             1             0
Disk Alarm                           0             1             0
Line Alarm                           0             0             0
Port Alarm                           0             0             0
Feeder Alarm                         0             0             0
Channel Alarm                        0             0             0
```

Execute **dspcds** to get an idea of the card state and then, if necessary, execute **dspcd** for slot 8 to acquire more details on the state of the card in slot 8. However, the output of **dspcds** shows that the card set is missing, which also explains the disk error.

```
ITNODE3.7.PXM.a > dspcds
ITNODE3                          System Rev: 02.00   Dec. 02, 2000 13:50:15 PST
Backplane Serial No: SAA03140750 Bp HW Rev:  B0      GMT Offset: -8
                                                     Node Alarm: MAJOR
Card  Front/Back       Card           Alarm      Redundant   Redundancy
Slot  Card State       Type           Status     Slot        Type
---   ----------       --------       --------   -------     -----

01    Active/Active    AXSM_16T3E3    NONE       NA          NO REDUNDANCY
02    Empty            ---            ---        ---         ---
03    Active/Active    AXSM_16OC3     NONE       NA          NO REDUNDANCY
04    Empty            ---            ---        ---         ---
05    Active/Active    AXSM_1OC48     NONE       NA          NO REDUNDANCY
06    Empty            ---            ---        ---         ---
07    Active/Active    PXM45          NONE       08          PRIMARY SLOT
08    Empty Resvd/Emp  ---            MAJOR      07          SECONDARY SLOT
09    Empty            ---            ---        ---         ---
10    Empty            ---            ---        ---         ---
11    Empty            ---            ---        ---         ---
12    Empty            ---            ---        ---         ---
13    Empty            ---            ---        ---         ---
14    Empty            ---            ---        ---         ---
```

dsppnportloscallrel

Display PNNI Port Loss of Signal Call Release

This command displays the enable status of the LOS call release feature. See **cnfpnportloscallrel** for a description of this feature.

Cards on Which This Command Runs

PXM45

Syntax

`dsppnportloscallrel` <*portid*>

Syntax Description

portid	The *portid* represents the PNNI logical port and has the format [*shelf.*]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.

Related Commands

cnfpnportloscallrel

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

First, confirm that LOS call release is disabled on port 3:1.1:1. Enable it, then confirm that it's enabled.

```
8850_NY.8.PXM.a > dsppnportloscallrel 3:1.1:1
```

Call release on Los :disabled

```
8850_NY.8.PXM.a > cnfpnportloscallrel 3:1.1:1 yes
```

```
8850_NY.8.PXM.a > dsppnportloscallrel 3:1.1:1
Call release on Los :enabled
```

dspportcnt

Display Port Counters

Displays ATM cell counters for a logical port. Refer to the example for contents.

Cards on Which This Command Runs

AXSM

AXSM Syntax

`dspportcnt` *`<ifNum>`*

Syntax Description

ifNum	Logical port number. The range depends on the card. For AXSM, the range 1–60.

Related Commands

dspports, **dspport**, **cnfport**, **dspcds**

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Examples

Display port counters on logical port (*ifNum*) 1 of the current AXSM.

```
MGX8850.1.AXSM.a > dspportcnt 1

Cleared at      : 06/27/2001 17:43:13
Current time    : 06/27/2001 17:44:43
Elapsed time    : 0 day(s) 0:1:30 [hh:mm:ss]

                          Total            Running Avg (cps)   Peak
 --------------------------------------------------------------------
 Arrival CLP0    Ing: 0000000000000000000000            0            0
 Arrival CLP1    Ing: 0000000000000000000000            0            0
 Ar CLP0 discard Ing: 0000000000000000000000            0            0
 Ar CLP1 discard Ing: 0000000000000000000000            0            0
 Departure CLP0  Ing: 0000000000000000000000            0            0
 Departure CLP1  Ing: 0000000000000000000000            0            0

 Arrival CLP0    Egr: 0000000000000000000000            0            0
 Arrival CLP1    Egr: 0000000000000000000000            0            0
 Ar CLP0 discard Egr: 0000000000000000000000            0            0
 Ar CLP1 discard Egr: 0000000000000000000000            0            0
 Departure CLP0  Egr: 0000000000000000000000            0            0
 Departure CLP1  Egr: 0000000000000000000000            0            0
```

dspprfhist

Display Performance History

The **dspprfhist** command shows the percentage of activity of tasks. Refer to the Example section for the type of displayed information. The command applies primarily to internal Cisco developers.

Cards on Which This Command Runs

PXM45, AXSM

Syntax

dspprfhist

Syntax Description

This command takes no parameters.

Attributes

Log: no log	State: active, standby	Privilege: ANYUSER

Example

Display the performance history on the current AXSM. The example display shows a completely idle system.

```
pop20two.1.AXSM.a > dspprfhist
CURRENT TIME         0:45:4
Sample #     0
0:44:27(From)-0:44:47(To)
TASK                 TaskId           %
---------------------------------------
INTERRUPT            -           0.0000
KERNEL               -           0.0000
IDLE                 -         100.0000
UNKOWN               -           0.0000

Sample #    -1
0:44:7(From)-0:44:27(To)
TASK                 TaskId           %
---------------------------------------
INTERRUPT            -           0.0000
KERNEL               -           0.0000
IDLE                 -         100.0000
UNKOWN               -           0.0000

Sample #    -2
0:43:47(From)-0:44:7(To)
TASK                 TaskId           %
---------------------------------------
```

dspswalm

Display Switching Alarms—displays alarms for switching circuits on the PXM45.

The **dspswalm** command can be used to determine if a card should be returned to Cisco for repair. For optional fault isolation after you execute **dspswalm**, use the crossbar commands to see if crossbar errors have occurred. The crossbar-specific commands apply to hardware or software developers who may need analyze the behavior of the switch ASIC. In the PXM45, if one switch ASIC (out of three) has serious errors, the total switch capacity falls short of the maximum 45 Gbps.

The following is a top-down sequence of alarm-related commands that lead to isolating possible faults in the switch fabric (or crossbar):

1. **dspndalms**
2. **dspswalms**
3. **dspxbaralm**
4. **dspxbarerrcnt**

An alarm can originate in any of the following:

- The backplane or any other part of signal path that connects a service module with a switch ASIC
- Crossbar ASIC on the PXM45
- Buffer circuitry on a service module

The definition of each alarm severity comes from Bellcore TR-NWT-000474. An alarm can be:

- *Critical*, indicating complete, non-recoverable failure, loss of data, and do on. The failed entity must be restored. A power failure or a line being disconnected is an example.
- *Major*, indicating service-affecting errors. This event indicates that a major service is damaged or lost, but the existing traffic is not affected.
- *Minor*, indicating non-service affecting errors or errors on a remote node. Corrective action is appropriate to prevent a serious fault from developing. An example is a fan failure, where no subscribers are immediately affected, but calamity could result if the situation persists. Note that an accumulation of lower-level alarms does equal a higher-level alarm.

Cards on Which This Command Runs

PXM45

Syntax

```
dspswalms
```

Syntax Description

This command takes no parameters.

Related Commands

dspndalms, dspxbarerrcnt

Attributes

Log: no log	State: active	Privilege: ANYUSER

Examples

Display switching alarms.

```
node_chi.8.PXM.a > dspswalms
Node Switching Alarm Summary

Card Crossbar              Critical    0   Major    0   Minor    0
Crossbar Fabric            Critical    0   Major    0   Minor    0
Humvee Alarm               Critical    0   Major    0   Minor    0
```

dsptrapmgr

Display Trap Manager

Display details about all existing trap managers. The **dsptrapmgr** output shows:

- IP address of each trap manager
- Port number on the connected work station
- Row status
- Read trap flag stats
- Next trap sequence number

Of these elements, the IP address and port number result from **addtrapmgr**.

Cards on Which This Command Runs

PXM45

Syntax

dsptrapmgr

Syntax Description

This command takes no parameters.

Related Commands

addtrapmgr, **deltrapmgr**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display trap managers.

```
node19.8.PXM.a > dsptrapmgr
     ipAddress     PortNum  RowStatus   ReadTrapFlag  NextTrapSeqNum
  ---------------  -------  ----------  ------------  --------------
  171.71.55.21      2500       Add           Off       0
  172.29.65.87      2500       Add           Off       348
  172.71.59.21      2500       Add           Off       0

  LastTrapSeqNum:     385
  NumOfValidEntries:  3
```

dspxbar

Display Crossbar—display the configuration of the crossbars

The **dspxbar** command displays general information about the configuration of a switch plane (or switching fabric or crossbar, also synonymous with switch ASIC). The configuration normally has fixed default values and is not configurable.

Note The low-level **dspxbar** command normally applies to software or hardware development and therefore is not useful for troubleshooting at the network or node level. Normally, to isolate a switching problem to determine whether to return a card, the **dspswalm** command is sufficient.

The crossbar-related commands are special debug commands with infrequent use. An engineer that needs to alter certain error thresholds for hardware or software development can use the **cnfxbarerrthresh** commands.

Connectivity Between the Switching Fabric and Card Slots

Before attempting to interpret the contents of the **dspxbar** output, some perspective on the crossbar information is appropriate. The focus of the crossbar commands is from the switch ASICs outwards, to the card slots. The reference point in relation to card slots is important and has to do with the wiring of the backplane. The connectivity between the cards and the switch fabric—consisting of all the switch ASICs in total—forms a mesh: each switch ASIC communicates with each card, and each card can use any ASIC. Therefore, if one ASIC becomes inoperative, the switching fabric continues to support new or existing connections within the switch, but the throughput falls far short of the maximum. Of less drastic effect would be a condition where one backplane trace can no longer carry data between an individual ASIC pin and a particular card slot. The **dspxbarerrcnt** command can help isolate such faults. Regardless, at the node level, the **dspswalms** command is sufficient for determining whether to replace the card with switching problem.

The Contents of the dspxbar Output

Refer to the examples to see the location of each of the following fields. The general information that **dspxbar** displays is the:

- Selected ASIC number (default 0 is unspecified).
- Number of the slot where the crossbar ASIC resides (7 or 8 for an MGX 8850 node).
- Number of the ASIC (0–2 in an MGX 8850 node).
- Revision number of the ASIC.
- Status of the ASIC.The status is either failed or OK. If the status is failed, the other ASICs must carry the switching load, and the throughput of the switch falls below the maximum. In this case, Cisco Systems recommends you replace the card.
- The cell grant mode is always Multicast Preferred.
- The Resync Sframe Tic is the rising edge of the clock. Sframe refers to a switch frame. (A switch frame is a 60-byte cell that carries a 53-byte ATM cell plus a special header for internal fabric use between the switching fabric and the service module.

The crossbar configuration consists of four categories of information for each slot:

- The type of backpressure is always Inband (meaning ingress direction). A crossbar does not have buffers (as the AXSMs do) and therefore must send backpressure signals to the queuing chips on each AXSM if congestion begins to occur in the switching plane.
- The Disable Request field automatically indicates any requests to turn off a source or destination for the link between the ASIC and the service module. The hardware alone generates the disable request, so you cannot make a disable request as part of troubleshooting.
- The Disable Data field indicates whether data transfer has been turned off for source or destination. In the current product, the field for source and destination always is No.
- The Redundancy Configuration field shows the mode of redundancy used by the ASIC and whether redundancy configuration exists for a slot. The redundancy mode is always Remap. Remap means that the switching fabric automatically maps cell transfers to the correct slot if a switchover occurs in a redundant pair. Note that, if a switchover occurs, the *logical* slot number in the endpoint ID stays the same.

  The Slot column for Redundancy Configuration shows whether card redundancy exists, as follows: if the slot number under Redundancy Configuration *differs* from the number in the Slot Number column (far left in the example screen), a card redundancy configuration exists for that slot pair.

Note From the number of fixed values in the preceding fields, you can see that very little can change in the crossbar configuration itself.

Cards on Which This Command Runs

PXM45

Syntax

dspxbar [*slot*] [*plane*]

Syntax Description

slot	(Optional.) The slot number of the XM60. Valid slot numbers are 9, 10, 25, and 26. Subsequently, specify a switching plane in the range 0 to 3.
plane	(Optional.) The number of the switching plane. If you do not specify a plane, the system displays information for plane 0 as a default. In an MGX 8850 node, the range is 0–2.

Related Commands

dspswalm, **dspxbaralm**, **dspxbarerrthresh**, **dspxbarerrcnt**, **clrxbaralm**, **clrxbarerrcnt**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display switch ASIC (or switch plane or crossbar) 0.

```
MGX8850.8.PXM.a > dspxbar
MGX8850                          System Rev: 02.00   Jul. 13, 2000 18:14:33 GMT

PXM45 CROSSBAR CONFIGURATION
Crossbar Slot No:  7         Switch Asic No:  0    Status: OK
Cell Grant Mode: Multicast Pref         Resync Sframe Tic: Rising-Edge Detect
Asic Revision: 1
Slot  BACK PRESSURE    DISABLE REQUEST  DISABLE DATA   REDUNDANCY CONFIG
 No   Grant   Mode       Dest    Src      Dest   Src     Mode         Slot
----  -------------    ---------------  ------------   ------------------
  1   Valid   InBand      No     No        No     No     Remap           1
  2   Valid   InBand      No     No        No     No     Remap           2
  3   Valid   InBand      No     No        No     No     Remap           3
  4   Valid   InBand      No     No        No     No     Remap           4
  5   Valid   InBand      No     No        No     No     Remap           5
  6   Valid   InBand      No     No        No     No     Remap           6
  7   Valid   InBand      No     No        No     No     Remap           7
  8   Valid   InBand      No     No        No     No     Remap           8
  9   Valid   InBand      No     No        No     No     Remap           9
 10   Valid   InBand      No     No        No     No     Remap          10
 11   Valid   InBand      No     No        No     No     Remap          11
 12   Valid   InBand      No     No        No     No     Remap          12
 13   Valid   InBand      No     No        No     No     Remap          13
 14   Valid   InBand      No     No        No     No     Remap          14
```

dspxbaralm

Display Crossbar Alarms—display severity of crossbar alarms.

The **dspxbaralm** command shows whether a crossbar alarm is minor, major, or critical. The display shows status on both the active and standby PXM45.

Note This low-level command normally applies to software or hardware development and therefore is not useful for troubleshooting at the network or node level. For troubleshooting at the switch level, use **dspndalms** and **dspswalms**.

For fault isolation after you execute **dspxbaralm,** you can survey the possible errors (illustrated in the example). The following is a top-down sequence of alarm-related commands:

1. **dspndalms**
2. **dspswalms**
3. **dspxbaralm**
4. **dspxbarerrcnt**

Cards on Which This Command Runs

PXM45

Syntax

dspxbaralm

Syntax Description

This command takes no parameters.

Attributes

Log: no log	State: active	Privilege: ANYUSER

Related Commands

dspxbar, **dspxbarerrcnt**, **dspxbarerrthresh**, **dspndalms**, **dspswalms**

Example

Display the severity of crossbar alarms. On the card in slot 7, crossbars 1 and 2 show errors. In such a case as two crossbars showing major errors, a card in a slot rather than a switch ASIC may be at fault. Therefore, you could execute **dspcds** to see if service module has an alarm or **dspxbarerrcnt** for switch ASICs 1 and 2 to see if each ASIC is showing similar data for the same card slot. You could also execute **dspcdalm** on the PXM45.

```
MGX8850.7.PXM.a > dspxbaralm
MGX8850                          System Rev: 02.00   Aug. 06, 2000 17:46:15 GMT
MGX8850                                              Node Alarm: CRITICAL
     Slot   Plane         Severity
     ----   -----         --------
        7       0             None
        7       1            Major
        7       2            Major
        8       0             None
        8       1             None
        8       2             None
```

```
Subsequent execution of dspxbarerrcnt shows the slot number (or link) and types of errors.
Note the large number of errors in slot 12. Many of the errors are parity or CRC related,
but the one error that is most serious is loss of sync. With a loss if sync between the
service module and the switching fabric, many other, comparatively less significant errors
are possible.
```

```
MGX8850.7.PXM.a > dspxbarerrcnt 7 2
MGX8850                          System Rev: 02.00   Aug. 06, 2000 17:47:53 GMT
MGX8850                                              Node Alarm: CRITICAL

                PXM45 CROSSBAR CURRENT ERROR COUNTERS

Crossbar Slot No: 7             Switch ASIC No: 2
        Loss    Rx      Rx      Xmit    Hdr     Pload   Slot    Slot    BP
Slot    Sync    Cv      Disp    Par     CRC     CRC     Remap   Recur   Par
------  ------  ------  ------  ------  ------  ------  ------  ------  ------
     1      0       0       0       0       0       0       0       0       0
     2      0       0       0       0       0       0       0       0       0
     3      0       0       0       0       0       0       0       0       0
     4      0       0       0       0       0       0       0       0       0
     5      0       0       0       0       0       0       0       0       0
     6      0       0       0       0       0       0       0       0       0
     7      0       0       0       0       0       0       0       0       0
     8      0       0       0       0       0       0       0       0       0
     9      0       0       0       0       0       0       0       0       0
    10      0       0       0       0       0       0       0       0       0
    11      0       0       0       0       0       0       0       0       0
    12   2772K  41594K  41594K      0   41594K  41594K      0       0   41594K
    13      0       0       0       0       0       0       0       0       0

MGX8850                          System Rev: 02.00   Aug. 06, 2000 17:47:53 GMT
MGX8850                                              Node Alarm: CRITICAL
    14      0       0       0       0       0       0       0       0       0

Frame Tick Error: 0
Frame Lock Error: 0
ACP Illegal Address Error: 0

Note: for big error counter, suffix K(x1000) or M(x1000000) is used.
```

dspxbarerrcnt

Display Crossbar Error Counters—display the count of various types of errors.

The **dspxbarerrcnt** command shows numbers of various types of errors on each slot-link. Note that the error can occur anywhere along the path of the ASIC and the hardware on the service module. The types of errors apply to the 60-byte switch frames.

Note This low-level command applies to software or hardware development and is not useful for troubleshooting the network or node. For troubleshooting the node, use **dspndalms** and **dspswalms**.

You can see the thresholds of the alarms that the errors trigger by using **dspxbarerrthresh**.

The following types and instances of errors appear by slot number (see example screen):

- Loss of synchronization between the ASIC and the queuing circuitry on the service module. The synchronization in this case applies to the timing of the internal switching frames (Sframes). Loss of synchronization is a very serious error.
- Receiver code violations (Rx Cv column in the display).
- Receiver disparity errors (Rx Disp column in the display). A disparity error is a summary of five ASIC-specific alarms.
- Transmitter parity errors.
- CRC failures for the header or the payload of the 60-byte Sframe.
- Failures to remap between slots as needed or excessive remapping between slots (Slot Remap and Slot Recur columns in the display).
- Parity errors in back-pressure messages.

A top-down sequence of troubleshooting commands for isolating faults in the switching fabric are:

1. **dspndalms**
2. **dspswalms**
3. **dspxbaralm**
4. **dspxbarerrcnt**

Cards on Which This Command Runs

PXM45

Syntax

`dspxbarerrcnt` *`<slot>`* *`<plane>`*

Syntax Description

slot	The slot of the switching fabric. On the MGX 8850 node, the slot is 7 or 8.
plane	The *plane* is the number of the switch ASIC. On the MGX 8850 node, the range for either slot is 0–2.

Related Commands

dspxbar, **dspxbaralm**, **dspxbarerrthresh**, **cnfxbarerrthresh**, **dspndalms**, **dspswalms**

Attributes

Log: log	State: active, standby	Privilege: ANYUSER

Example

Display the crossbar error counters for switch ASIC 2 in slot 7. The Node Alarm field of the display shows the errors have resulted in a critical alarm. A large number of errors have occurred in the slot 12 link. Many of the errors are parity or CRC related, but the one error that is most serious is loss of sync. With a loss if sync between the service module and the switching fabric, many other, comparatively less significant errors are possible.

Note at the bottom of the display a message about multipliers for large numbers of errors. Because the display can accommodate a finite number of errors without being distorted, a K (for 1000) and M (for 1000000) multiplier may be necessary, as this example shows.

```
MGX8850.7.PXM.a > dspxbarerrcnt 7 2
MGX8850                          System Rev: 02.00   Aug. 06, 2000 17:47:53 GMT
MGX8850                                              Node Alarm: CRITICAL

                PXM45 CROSSBAR CURRENT ERROR COUNTERS

Crossbar Slot No: 7             Switch ASIC No: 2
        Loss    Rx      Rx      Xmit    Hdr     Pload   Slot    Slot    BP
Slot    Sync    Cv      Disp    Par     CRC     CRC     Remap   Recur   Par
------  ------  ------  ------  ------  ------  ------  ------  ------  ------
     1       0       0       0       0       0       0       0       0       0
     2       0       0       0       0       0       0       0       0       0
     3       0       0       0       0       0       0       0       0       0
     4       0       0       0       0       0       0       0       0       0
     5       0       0       0       0       0       0       0       0       0
     6       0       0       0       0       0       0       0       0       0
     7       0       0       0       0       0       0       0       0       0
     8       0       0       0       0       0       0       0       0       0
     9       0       0       0       0       0       0       0       0       0
    10       0       0       0       0       0       0       0       0       0
    11       0       0       0       0       0       0       0       0       0
    12   2772K  41594K  41594K       0  41594K  41594K       0       0  41594K
    13       0       0       0       0       0       0       0       0       0

MGX8850                          System Rev: 02.00   Aug. 06, 2000 17:47:53 GMT
MGX8850                                              Node Alarm: CRITICAL
    14       0       0       0       0       0       0       0       0       0
Frame Tick Error: 0
Frame Lock Error: 0
ACP Illegal Address Error: 0
Note: for big error counter, suffix K(x1000) or M(x1000000) is used.
```

dspxbarerrthresh

Display Crossbar Error Threshold—display the thresholds for alarms of different severities.

A crossbar can have nine types of errors, and each error has a threshold. The errors are loss of synchronization, a variety of parity and CRC errors, and so on (see Definitions of Crossbar Errors).

The items that make up a threshold are the:

- Duration of the errored state
- Number of errors during that time period
- Upper and lower error counts within a particular alarm severity (minor, major, and critical)

Note The default settings for crossbar error thresholds are optimal for nearly all applications. The **dspxbarerrthresh** command shows the existing thresholds. If necessary, you can change thresholds through the **cnfxbarerrthresh** command.

The two types of alarm counts for each of these severities. (Refer to the example.) The higher count is the Alarm Count and is the highest number of errors that triggers an alarm of a particular severity. The lower count is the Release Count: when the number of errors drops below the Release Count, the alarm severity drops to the next lower severity. For example (using the defaults shown in the example display), if the number of transceiver errors drops below 40 (a major alarm), the alarm turns into a minor alarm. The higher count for a minor, major, and critical alarm is the number of errors that trigger that alarm. The lower count is number of errors that causes the severity to drop to the next lower severity.

The types of errors whose thresholds are displayed are:

1. Loss of synchronization (LossOfSync)
2. Transceiver error (TranscieverErr)
3. DisparityErr—an accumulation of five ASIC-level errors
4. ParityErr—a parity error in the switch frame as a whole
5. HeaderCRCErr—a CRC error for the switch frame header
6. PayloadCRCErr—a CRC error for the switch frame payload
7. RemapTwiceErr
8. RemapRecurrErr
9. Backpressure parity error (B.P.ParityErr)—a parity error in the signaling for backpressure

Cards on Which This Command Runs

PXM45

Syntax

dspxbarerrthresh

Syntax Description

This command takes no parameters.

Related Commands

dspxbar, **dspxbaralm**, **dspxbarerrcnt**

Attributes

Log: no log	State: active	Privilege: ANYUSER

Example

Display the current crossbar error thresholds.

```
JBP2_Lower.8.PXM.a > dspxbarerrthresh
JBP2_Lower                     System Rev: 02.01   Nov. 28, 2000 21:39:29 GMT
MGX8850 (JBP-2)                                    Node Alarm: MAJOR
                            CROSSBAR ERROR CONFIGURATION
                     Thresh  -- MINOR --     -- MAJOR --     -- CRITICAL --
Device Error         Time    Clear Alarm     Clear Alarm      Clear Alarm
Type                 (msec)  Count Count     Count Count      Count Count
-------------------- ------  ----- -----     ----- -----      ----- -----
LossOfSync            20000      0     3         4    15        300   301
TranscieverErr        20000      0    31        40   150        300   301
DisparityErr          20000      0    31        40   150        300   301
ParityErr             20000    300   301       300   301        300   301
HeaderCRCErr          20000      0    31        40   150        300   301
PayloadCRCErr         20000      0    31        40   150        300   301
RemapTwiceErr         20000      0     1         0     1        300   301
RemapRecurrErr        20000    300   301       300   301        300   301
B.P.ParityErr         20000      0    31        40   150        300   301
```

dspxbarmgmt

Display Crossbar Management—display the load sharing configuration.

The **dspxbarmgmt** command shows details about the load sharing configuration for the node. See the **cnfxbarmgmt** description for an explanation.

Cards on Which This Command Runs

PXM45

Syntax

```
dspxbarmgmt
```

Syntax Description

This command takes no parameters.

Related Commands

cnfxbarmgmt

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the crossbar management state for the current node. The settings are the defaults.

```
pop20two.7.PXM.a > dspxbarmgmt
pop20two                         System Rev: 02.01   Dec. 05, 2000 20:43:43 GMT
MGX8850                                              Node Alarm: MAJOR
Load Sharing: Enable
Auto Shutdown: Disable
Plane Alarm Threshold: 3
```

dspxbarstatus

Display Crossbar Status

Display status of each slot for a crossbar.

Cards on Which This Command Runs

PXM45

Syntax

dspxbarstatus [*plane*]

Syntax Description

plane	(Optional.) The number of the switching plane. The default is 0. For the MGX 8850 node, the range is 0–2.

Related Commands

dspxbar, **dspxbaralm**, **dsperrcnt**, **dsperrthresh**

Attributes

Log: no log	State: active, standby, init	Privilege: ANYUSER

Example

Display the status of switch ASIC 0. The active PXM45 in this example is in slot 7.

```
MGX8850                          System Rev: 02.00   Jul. 13, 2000 18:26:54 GMT
MGX8850                                              Node Alarm: UNKNOWN
Switch CD No: 7          Switch ASIC No: 0
Administrative Status Bitmap: 0xFFFFFFFF
Operational Status Bitmap   : 0x0
Error Status Bitmap:
Slot 1: 0x0
Slot 2: 0x0
Slot 3: 0x0
Slot 4: 0x0
Slot 5: 0x0
Slot 6: 0x0
Slot 7: 0x0
Slot 8: 0x0
Slot 9: 0x0
Slot 10: 0x0
Slot 11: 0x0
Slot 12: 0x0
Slot 13: 0x0
Slot 14: 0x0
Slot 15: 0x0
Slot 16: 0x0
```

```
Misc Error Bitmap: 0x0
```

dumptrace

Dump Trace—place the contents of a trace in a log file and show the name of the file.

The output of **dumptrace** is a filename. Provide this name to **dsplog -log** option to see its contents.

Cards on Which This Command Runs

PXM45

Syntax

`dumptrace`

Syntax Description

This command takes no parameters.

Related Commands

dsplog, **dsplogs**

Attributes

Log: no log	State: active, standby	Privilege: SERVICE_GP

Example

Execute **dumptrace**, then view the contents with **dsplog**.

```
pop20one.7.PXM.a > dumptrace
The trace is saved in file error50.log
```

Display the contents of error log file 50. This example shows only the first screen.

```
pop20two.7.PXM.a > dsplog -log error50
07-00140 11/30/2000-23:10:59  SCM-4-RESP_TIMEOUT
         tSCM       0x80260d20
 Timeout waiting for response for slot 9, report to PMM
07-00139 11/30/2000-23:10:20  CLI-7-CMDLOG
         tDbgCmdTsk 0x80377d9c
 cliCmdLog: cisco@console: "cnfname pop20two".
07-00138 11/30/2000-23:10:20 PROO-6-soPRctcEvRcv
         PnProot    0x8068b894
 proot_ctc_event_hdlr(): received EVENT 0x63 from CTC
07-00137 11/30/2000-23:10:08  CLI-7-CMDLOG
         tDbgInTask 0x80377d9c
 cliCmdLog: cisco@console: (cc 7).
07-00136 11/30/2000-23:10:07  CLI-7-CMDLOG
         tDbgInTask 0x80377d9c
 cliCmdLog: cisco@console: (login).
07-00134 11/30/2000-23:08:23 SHM_-7-CARD_FAIL
         ShelfMgr   0x802e73b4
 SHM: slot 9 failed reason SHM_CDF_MAX_RESETS_REACHED [7]
07-00135 11/30/2000-23:08:23 SHM_-7-ALM_SET
         ShelfMgr   0x8030ccf4
 SHM Alarm Raised: pslot: 9, AlarmId: 0x2800b(SHM_ALM_NON_RES_FC_FAILED) - 3 dropped
```

offdiagstat

Off Diagnostics Statistics

Halts the statistical diagnostic program that keeps count of how many times the diagnostics have run.

Syntax

`offdiagstat`

Syntax Description

This command takes no parameters.

Cards on Which This Command Runs

AXSM

Related Commands

ondiagstat

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

```
pop20one.10.AXSM.a > offdiagstat

Disabling diag stats, enabling bucket stats.
```

ondiagstat

On Diagnostics Statistics

Starts running the diagnostics statistics program that keeps count of how many times diagnostics has run.

Syntax

`ondiagstat`

Syntax Description

This command takes no parameters.

Cards on Which This Command Runs

AXSM

Related Commands

offdiagstat

Attributes

Log: log	State: active	Privilege: SERVICE_GP

Example

```
pop20one.10.AXSM.a > ondiagstat

Enabling diag stats, disabling bucket stats.
```

pathtraceie

Path Trace IE

Removes or inserts path trace information element (IE) at port level.

Cards on Which This Command Runs

PXM45

Syntax

```
pathtraceie <portid>
{rmv | ins}
```

Syntax Description

portid	The *portid* represents the PNNI logical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.
rmv	Allow to remove Trace Transit List IE at the port.
ins	Allow to insert Trace Transit List IE at the port.

Related Commands

conntrace, **pathtraceport**, **pathracenode**

Attributes

Log: log	State: active	Privilege: SUPER_GP

pathtracenode

Path Trace Node

Enables or disables path trace feature at the node level.

Cards on Which This Command Runs

PXM45

Syntax

pathtracenode {enable | disable}

Syntax Description

enable/disable	Enables (disables) path trace at node level.

Related Commands

conntrace, pathtraceport, pathtraceie

Attributes

Log: log	State: active	Privilege: SUPER_GP

pathtraceport

Call Control Operations: Enable/Disable Path Trace at Port Level

Enables (disables) path trace feature at port level, then saves the result of the path trace in the log file.

Cards on Which This Command Runs

PXM45

Syntax

```
pathtraceport <portid>
{enable | disable}
[-H {on | off}]
[-CB {on | off}]
[-V {on | off}]
[-CR {on | off}]
[-cldnum called-AESA]
[-clgnum calling-AESA]
```

Syntax Description

portid	The *portid* represents the PNNI logical port and has the format [*shelf*.]*slot*[:*subslot*].*port*[:*subport*]. See also PNNI Format, page 9-3.
enable | disable	Enables (disables) path trace at port level.
-H	Specifies the hierarchy option. If enabled, information from all the DTLs in the hierarchy are added in the TTL IE. Default = off
-CB	Specifies the crankback option. If enabled, the failure cause for crankback is included in the TTL IE. Default = on
-V	Specifies the VPI/VCI option. If enabled, VPI/VCI values of the egress port are added in the TTL IE at every node. Default = off
-CR	Specifies the call reference option. If enabled, call reference values of all egress ports are added in the TTL IE. Default = off
-cldnum	Specifies the called party number. Enables (disables) path trace on a specific called address.
-clgnum	Specifies the calling party number. Enables (disables) path trace on a specific calling address.

Related Commands

conntrace, pathracenode, pathraceie

Attributes

Log: log State: active Privilege: SUPER_GP

Examples

Format of the path trace result for a call in the log file:

```
Result:Succ/Fail  Reason: "Desc"  Called No: "--"  Calling No: "--"
Cause: "value:Desc"
NodeId  Ingress Port  Egress Port  Vpi/Vci  Call Ref  EndPtRef (opt)
XXXX     YYYY          ZZZZ         aaa/bbb    cccc       dddd
XXXX     YYYY          ZZZZ         aaa/bbb    cccc       dddd
```

resetcd

Reset Card—resets the card or the failure history.

The **resetcd** command lets you reset a card or, if you specify the appropriate option, the failure history of a card.

Using the **resetcd** command without defining a slot number results in an ungraceful (disruptive) upgrade. This is the fastest method to upgrade a card, but it disrupts service.

A graceful (non-disruptive) upgrade requires that the **loadrev**, **runrev**, and **commitrev** commands have already been executed and that the card is identified in the command string.

Cards on Which This Command Runs

PXM45

Syntax

```
resetcd
[slot]
[ -f]
```

Syntax Description

slot	The optional *slot* parameter identifies a card to reset. If you do not specify a slot, the command targets the current PXM45.
-f	Specifies that the command resets only the failure history of a card.

Related Commands

resetsys

Attributes

Log: no log	State: active, init	Privilege: SUPER_GP

Example

Reset the current PXM45.

```
pinnacle.7.PXM.a > resetcd
The card in slot number 7, will be reset. Please confirm action
Do you want to proceed (Yes/No)? n
(command not executed)
```

Reset the failure history of the card in slot 1.

```
8850_NY.7.PXM.a > resetcd 1 -f
Clearing (-F) Failed state on slot 1. Please confirm action
resetcd: Do you want to proceed (Yes/No)?
```

resetsys

Reset System

Reset the entire node.

Cards on Which This Command Runs

PXM45

Syntax

```
resetsys
```

Syntax Description

This command takes no parameters but displays a warning and prompts you to continue the execution.

Related Commands

resetcd

Attributes

Log: no log	State: active, init	Privilege: SUPER_GP

Example

Reset the system.

```
pinnacle.7.PXM.a > resetsys
This command resets the entire shelf, a destructive command.
Please confirm now!
Do you want to proceed (Yes/No)? n
(command not executed)
```

switchcc

Switch Core Cards

Switch control of the node from the active PXM45 to the standby PXM45. If a standby PXM45 is not available, the **switchcc** command fails.

You cannot execute **switchcc** during a configuration-copy. If you attempt it, the system blocks the command. Furthermore, the command fails if the switch is unstable or if the standby PXM45 is not in the standby state.

Cards on Which This Command Runs

PXM45

Syntax

switchcc

Related Commands

None

Attributes

Log: no log	State: active	Privilege: SERVICE_GP

Example

Attempt a **switchcc** without a standby PXM45 in the backplane.

```
raviraj.7.PXM.a > switchcc
Do you want to proceed (Yes/No)? y

 Core card redundancy unavailable

raviraj.7.PXM.a >
```

tstconseg

Test Connection Segment

Test the integrity of an SVC or SPVC. With **tstconseg**, a single collection of supervisory cells is sent in the *egress* direction between the card and service equipment (CPE). (See **tstdelay** for *ingress* direction.)

When the test successfully starts, the system displays a message stating that the test has begun and directs you to use either **dspcon** or **dspchantests** to see the results. The **dspcon** command shows detailed information about the connection and has a field for the results of this test. The **dspchantests** command display only the results of the test.

Note The **dspcon** fields on the AXSM for round trip delay—including the status of OAM loopback—always show the results of the latest test and are not changed until a new execution of **tstconseg** or **tstdelay**. Therefore, re-executing **dspcon** does not clear the value for RTD or the indication that an OAM loopback is present. The only way to reset these fields to null is to down the port (through **dnport**).

Cards on Which This Command Runs

AXSM

Syntax

```
tstconseg <ifNum> <vpi> <vci>
[-num <iterations>]
```

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	The VPI range for the SVC or SPVC is 1–255.
vci	The VCI range for the SVC is 32–65535.
-num	(Optional) Specifies the number of times a collection of supervisory cells should traverse the SVC for the current execution of **tstconseg**.

Related Commands

dspcon, **tstdelay**, **dspchantests**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Test the integrity of 1 10 1000 in the egress direction.

```
node19.1.AXSM.a > tstconseg 1 10 100
Test started; Use dspcon/dspchantests to see test results
```

tstdelay

Test Delay

Test the integrity of the connection in the ingress direction by sending a collection of supervisory cells to the remote end of the network and back. (See **tstconseg** for the egress direction.) The **tstdelay** command applies to only SPVCs.

If the test successfully begins, the display states the fact and directs you to use the **dspcon** or **dspchantests** command to view the round trip time in microseconds. The **dspcon** display shows detailed information on the connection and has a field for the test results. The **dspchantests** display shows the results of only the round trip delay test.

Note The **dspcon** fields on the AXSM for round trip delay—including the status of OAM loopback—always show the results of the latest test and are not changed until a new execution of **tstconseg** or **tstdelay**. Therefore, re-executing **dspcon** does not clear the value for RTD or the indication that an OAM loopback is present. The only way to reset these fields to null is to down the port (through **dnport**).

Note The primary purpose of **tstdelay** is to test the integrity of the connection. The round trip time is not accurate enough for any use that requires an accurate measurement of delay.

Cards on Which This Command Runs

AXSM

Syntax

```
tstdelay <ifNum> <vpi> <vci>
[-num <iterations>]
```

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	Virtual path identifier. On the AXSM, the range is 1–255.
vci	Virtual connection identifier. On the AXSM, the range is 32–65535 for a VCC. For a VPC, the *vci* is 0.
-num	(Optional) Specifies the number of times a collection of supervisory cells should traverse the SVC for the current execution of **tstdelay**.

Related Commands

dspcons, **tstconseg**, **dspcon**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

On the AXSM slot, get the round-trip delay for connection 1 10 100. This example contains four command executions to illustrate how to obtain a list of logical ports; obtain a connection number; start the test; and view the results. the commands are **dspports**, **dspcons**, **tstdelay**, and **dspcon**.

Step 1 Identify the logical ports on the card by executing **dspports**. For this example, the logical port (ifNum in the display) is 1.

```
node19.1.AXSM.a > dspports
ifNum Line Admin Oper. Guaranteed Maximum     Port SCT Id     ifType  VPI
           State State Rate       Rate                               (VNNI only)
----- ---- ----- ----- ---------- --------- ----------------- ------ ----------
    1  1.1    Up    Up    1412831   1412831   6                  UNI    0
```

Step 2 Get the connection ID to provide to **tstdelay**. The connection identifier appears in NSAP format. In this example, assume **tstdelay** execution occurs at the slave end of the SPVC. Take the significant digits from the Identifier (01.0010.00100) to get the logical port, VPI, and VCI for **tstdelay**. These values are 1, 10, and 100.

```
node19.1.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld    Alarm
------    ----------   ----   --------   ---    ----    -----
    0  01.0010.00100   VCC        ubr1   S    0000ebfb      none
    1  01.0011.00101   VCC        ubr1   M    0000ec27      none
```

Step 3 Execute **tstdelay** for logical port 1, vpi 10, vci 100. The system response shows that the command started correctly and directs you to use **dspcon** or **dspchantests** to see the results.

```
node19.1.AXSM.a > tstdelay 1 10 100
Test started; Use dspcon/dspchantests to see test results
```

Step 4 Execute **dspchantests** to see the results as displayed by this command. The units of measure for the round trip delay is microseconds.

```
node19.1.AXSM.a > dspchantests 1 10 100
Connection Id    Test Type    Direction    Result     Round Trip Delay
=============    =========    =========    =======    ================
01.0010.00100:    OAM Lpbk     ingress     Success          30000
```

Step 5 Execute **dspcon** to see the results as displayed by this command. The line with test results appears towards the end of the display and begins with Loopback Type. The Direction field shows ingress, indicating the **tstdelay** command produced these results. (If **tstconseg** had been the last test command, this field would say egress.) The RTD (round trip delay) field shows 30000 microseconds.

```
node19.1.AXSM.a > dspcon 1 10 100
---------------------------------------------------------------------------
Local   :                NSAP  Address              port     vpi    vci
(S)       47009181000000000001A53C82D00000101180100  1.01.01     10     100
Remote  :                NSAP  Address              port     vpi    vci
(M)       47009181000000000001A53C82D00000101180100  1.01.01     11     101
---------------------------------------------------------------------------
Conn. Type    :     VCC                          Admn Status  : ADMN-UP
Service Type  :    ubr1                          Rtng Status  : -67372037
Controller    :       2
```

```
--------------------------------------------------------------------------
Local PCR     :      14                          Remote PCR   :      14
Local SCR     :       3                          Remote SCR   :       3
Local CDV     :      -1                          Remote CDV   :      -1
Local CTD     :      -1                          Remote CTD   :      -1
Local MBS     :       1                          Remote MBS   :       1
Local CDVT    :      -1                          Remote CDVT  :      -1
Admin weight  :      -1                          Frame discard:       N
--------------------------------------------------------------------------
OAM CC Config : DISABLED                         Statistics   : DISABLED
--------------------------------------------------------------------------
Loopback Type : OAM Lpbk | Dir: ingress | Status: Success | RTD:   30000 us
--------------------------------------------------------------------------

--------------------------------------------------------------------------
Port side Tx  :  normal                          Swth side Tx :  normal
Port side Rx  :  normal                          Swth side Rx :  normal
--------------------------------------------------------------------------
I-AIS/RDI   E-AIS/RDI   CONDITIONED    CCFAIL       IfFail      Mismatch
   NO           NO           NO           NO           NO          NO
--------------------------------------------------------------------------
```

upcon

Up Connection—return a downed connection to the active (administratively up) state.

Activate a connection that was previously brought down by the **dncon** command. (The typical purpose of **dncon** is some form of operational modification or troubleshooting.) Execute **upcon** at the master end of the connection.

Cards on Which This Command Runs

AXSM

Syntax

`upcon <ifNum > <vpi> <vci>`

Syntax Description

ifNum	The logical port number. The range for AXSM is 1–60.
vpi	Virtual path identifier. On the AXSM, the range is 1–255.
vci	Virtual connection identifier. On t he AXSM, the range is 32–65535 for a VCC. For a VPC, the only *vci* is 0.

Related Commands

dncon

Attributes

Log: log	State: active	Privilege: GROUP1

upln

Up Line

Activates a line on the current card. After you have activated the line, use **cnfln** to configure the line characteristics such as the type of line (for SONET types, T3, or E3), line signaling, and so on.

Note See description of **cnfcdsct** for important planning considerations before you use **upln**.

Cards on Which This Command Runs

AXSM

Syntax

`upln` *`<bay.line>`*

Syntax Description

bay.line	Identifies the bay (1 or 2) and the number of the line. The line number can be 1 to the highest numbered line on the back card.

Related Commands

dspln, **dsplns**, **cnfln**, **dnln**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Activate line 1 in bay 1.

```
MGX8850.1.AXSM.a > upln 1.1
```

upport

Up Port

The **upport** command returns a logical port to the up state (or ups the port) so the port can again carry traffic. The **upport** command concludes possible re-configuration or troubleshooting steps. Before you execute **upport**, you must have downed the port by executing **dnport**. Throughout the sequence of downing and upping a port, the configuration for the port remains intact whether the logical port is a UNI or an NNI.

The routes for connections vary by interface type:

- After you re-enable an NNI port through **upport**, you cannot return the re-routed connections to the upped port. The PXM45 routes connections over the trunk as needed.
- On a UNI, the connections continue to exist but remain in the failed state until you enable the port by executing **upport**.

Cards on Which This Command Runs

AXSM

Syntax

`upport <ifNum>`

Syntax Description

ifNum	A logical port (interface) number. Only one logical port is allowed if the line operates as a UNI or NNI. For the virtual network to network interface (VNNI), multiple ports can exist on a line. For AXSM, the range 1–60. Use **dspports** or **dspport** as needed to determine which port to bring up.

Related Commands

dspport, **dspports**, **dnport**

Attributes

Log: log	State: active	Privilege: GROUP1

Example

Restore port 1 on the current card to operation.

```
MGX8850.1.AXSM.a > upport 1
```

INDEX

Symbols

A

B

C

D

E

F

H

I

L

M

N

O

P

R

S

T

U

W